

OTDK-DOLGOZAT

2015

INFLATION IN A FANTASY ECONOMY

Kézirat lezárva: 2013. március 25.

TABLE OF CONTENTS

1. Introduction.....	1
2. Game Mechanics	4
2.1. The world of RuneScape	4
2.2. Real money trading.....	4
2.3. Grand Exchange	4
3. Literature Review	6
3.1. Money supply and demand.....	6
3.2. The effect of real money trading	7
3.3. The model	7
4. Research Focus.....	9
5. Items.....	11
5.1. Game activities	11
5.2. Consumption.....	12
5.3. Chosen items.....	13
6. Events to Consider	15
6.1. Evolution of combat	15
6.2. Bot-Free and bot-infested times	15
6.3. Player-versus-player combat changes	16
6.4. Bonus XP Weekends	16
7. The Lasperyes Index.....	18
7.1. Prices	18
7.2. Lasperyes weights.....	18
7.3. Results	20
8. The Paasche Index	22
8.1. Paasche weights.....	22
8.2. Results	23
9. Conclusion	25

Works Cited.....	26
Appendix.....	27

1. INTRODUCTION

Inflation is a well-known phenomenon in the real world as well as in the virtual economies of massively multiplayer online role-playing games (MMORPGs). The Fisher effect teaches us that there is a direct link between the nominal interest rate and inflation. However, the banks of MMORPGs do not tend to offer interest on deposits and yet inflation exists and is widely talked about. My goal is to determine what causes inflation in a game that has no interest rates and no central bank to issue money. Without the strong influence of the nominal interest rate I get a unique opportunity to see how other macroeconomic factors influence the price level.

Inflation is the general and steady increase of the price level. What this definition does not tell us are the reasons behind it. According to macroeconomic models, inflation can be caused by the increase of money supply, the increased real demand or shortage of goods or the velocity of money supply. Moreover, in an economy where nominal wages are sticky downwards, inflation is also the way of adjusting real wages to a lower level.

All of these factors differ from their real-life equivalents, some more than others. First of all, the money supply of MMORPGs is unique. In the real world, the money supply is determined by a central bank. In the virtual economy of these games, money constantly enters through sources called faucets and leaves through drains. The money circulating in the economy can be considered the money supply.

Like central banks, nominal rigidity is also an unknown concept. The price of goods quickly adjusts to supply and demand. In most games, players make money by producing items of their choice. If the price of the given item plummets, so does the player's wage. Obviously, players with more skills are less exposed to price shifts, since they can simply decide to collect something else.

Not only can prices change rapidly, so can the demand and supply of goods. Games tend to receive updates and extensions to keep old players from getting bored. For instance, introducing brand new items can make the old ones less desirable and modify demand. This phenomenon is also known in real life. Technology advances rapidly and we have access to more and more electrical gadgets every day. However, virtual economies do not have so many types of goods, therefore every single change is more influential and easier tracked.

To see how a system like this works, I decided to analyse the price level of a specific MMORPG, RuneScape, developed by Jagex Ltd. I chose this game because its central exchange

system allows me to access price data I could not acquire in other games. Therefore, in this game I am not only able to form hypotheses, I can also test them.

To determine whether the prices in RuneScape show inflationary or deflationary tendencies, I calculate a consumer price index for a period of 4 years. In order to get more accurate results, besides calculating the traditional Lasperyes version, I also try to estimate a Paasche version. My indices indicate that there are different periods in the game's life, some of which show inflation and others demonstrate deflation. RuneScape is a game that receives weekly updates and one update tends to bring more than one change. All these updates have the potential to influence the economy. While some effects are minor, others are so substantial that they can be seen in my consumer price index.

We can observe breakpoints, which imply that one or more factors have been altered. The best way to find the reasons behind the breaks is to check what updates happened around that time. I have found four major changes, all of which are related to the supply or demand for goods. This result is surprising, because most papers that mention inflation emphasise the importance of the faucet-drain system that determines the money supply. In real life, economists also tend to be concerned about the money supply, because this is the factor that can change the most rapidly and is usually the reason behind hyperinflation. What my index shows, however, is that in virtual economies the changes in the market of goods can be so substantial that they greatly affect the price level. At the same time, the effects of money supply modifications are small enough to get lost in the volatility of the index.

Another relevant topic popular with researchers is the effect of real money trading on the price level. RMT means selling virtual goods or currencies for real money. It can be done by individual players or companies who program third-party software or macros for this purpose. There are different theories on its effects on the price level. What my index shows is that in periods where macros are ubiquitous the price level is lower. The reason behind this phenomenon is that there are only a few ways of obtaining RuneScape's currency, the gold piece. Despite its name, the gold piece is fiat money, it has no intrinsic value. It has no relation to the gold ores and gold bars in the game, therefore even mining gold is not a direct way of earning money. Those operating a third-party software can either acquire gold pieces through one of the six faucets identified by Bilir (2009) or they can collect virtual goods and sell them for virtual money. The latter seems to be the more popular choice. Macros gather high amounts of items, usually raw materials, sell them and consequently push the prices down. In conclusion, real money trading causes deflation in the world of RuneScape.

This is not necessarily true for all MMORPGs. In a world where macros have comparative advantage in directly producing gold instead of items, they are likely to cause inflation. What's true in all cases is that real money trading devalues something. When macros produce resources which devalue these items, we are talking about deflation. When they directly obtain money which devalues the currency of the game, we are talking about inflation.

The third thing I wanted to find out was whether mudflation has a significant effect on the price level. Mudflation describes the phenomenon of virtual items becoming cheaper as they are rendered useless by new items brought in by regular expansions or updates. My Lasperyes index works with a fixed basket and I did not take out old or put in new items. Therefore, should mudflation be significant, we would be able to see it in the index. However, constant deflation cannot be observed.

This does not mean that mudflation does not affect certain items, but it definitely does not affect *all* of them. This was to be expected, since Jagex is very careful with introducing new items into the game. The new item is usually not superior to the old one in every single aspect. For example, the godswords released in 2007 are more powerful than the old whips, but they are slower and two-handed, whereas the whips are one-handed. For this reason, players not willing to sacrifice speed or the protection of a shield still choose to fight with a whip.

This paper has three important conclusions. First, when talking about inflation in a virtual economy, it is not enough to worry about the money supply, we also have to look at the demand for money. Second, if we want to determine whether macros cause inflation or deflation, we have to look at the unique specialties of the game. Everything depends on the players' comparative advantages. Third, the constant game updates to keep old players active do not have to result in significant mudflation.

2. GAME MECHANICS

I do not assume that all readers of this paper spend their free time playing RuneScape. On account of this, I dedicate a chapter to introducing the game, especially those aspects that are relevant for this research.

2.1. THE WORLD OF RUNESCAPE

Massively multiplayer online role-playing games are an immensely popular pastime. Millions of players choose to create avatars in these virtual worlds. The world of RuneScape is a fantasy-medieval world where players fight with swords, create magic potions and hunt mythical creatures. They can possess items and store them in a bank. They are also able to sell some of these items to other players, creating an economy.

Naturally, this economy is different from the real one in many aspects. There are less than 5000 types of tradable items, there are no companies and no government spending. It also changes more rapidly. An update implemented in less than 5 minutes can immediately modify the supply or demand for goods.

2.2. REAL MONEY TRADING

The process of selling in-game gold for real money is called real money trading or real world trading. In most games, this is against the rules, but programmers still struggle with preventing it.

There are two main categories of real money trading. The first is actual players selling gold to their friends or auctioning off their items when they quit the game. A few players doing this does not really affect the economy. The problem is with the second category. There are companies who use cheap labour or programs, the so-called macros or bots, to massproduce game gold. Theoretically, this could lead to an increased money supply which causes inflation in the game.

2.3. GRAND EXCHANGE

If a player wants to sell an item he no longer needs, he can walk into the city of Varrock and put an offer in the Grand Exchange. If there is someone who is willing to buy the aforementioned item for the given price, the system automatically connects the offers. Members can place up to six offers at a time and are able to keep playing or even go offline while they are waiting for the offer to be completed. Thanks to this efficient system, transaction costs are

practically zero, prices are flexible and they reflect supply and demand. Without the Grand Exchange, I would not be able to measure the game's inflation or, in fact, build any model involving item prices.

I do have to point out that the Grand Exchange did not always work the way it does today. Between 2 January 2008 and 1 February 2011, unbalanced trade was banned from the game. This meant that players could only offer money within a $\pm 5\%$ range of the item's price. However, if the initial price determined by the designers was far from the item's actual value, the item might never be traded on the Grand Exchange and the price got stuck on the wrong level. Another anomaly was that all goods had a minimum price under which they could not be sold/bought. If a useless item got stuck at this minimum, it was considered a so-called 'junk' item and was frequently used for unbalanced trade outside the Grand Exchange. Despite these two malfunctions, the Grand Exchange worked perfectly in case of most goods. Therefore I could measure the inflation even in the era of balanced trade, I just had to be careful not to pick the wrong items.

3. LITERATURE REVIEW

So far, there have been two types of studies on game inflation. The first category looks at the money market in general. It tries to find out how the money supply is composed in an economy without a central bank and what can cause inflation in this system. The second category focuses on a particular problem: the inflationary effect of real money trading. The reason why this problem is so popular with researchers is that it affects most MMORPGs, whereas others tend to be game-specific.

3.1. MONEY SUPPLY AND DEMAND

In this subchapter I briefly introduce previous analyses of the demand for money and the money supply. I also mention a previous attempt of measuring inflation using a price index.

Yamaguchi (2004) claims that that by choosing between saving and consumption economic agents influence the balance between money and goods. He also points out that, in a world without interest rates, economic agents are less motivated to save money.

Breau (2002) and Lewis (2003) use the concept of a faucet system to describe the ingame money supply. Money comes into the economy through faucets, such as NPC missions, NPC purchases and cash loot. It circulates within the sink and then leaves through drains like NPC consumables, housing maintenance or character deletion. Both of them claim that drains are less flexible than the faucets, which are usually controlled by players. A system like this is hard to balance and can easily end up in a hyper-inflated state.

Füleki et al. (2008) also support the model of faucets and drains. They state that several tax-like measures are implemented into MMORPGs to work as drains and not to finance the state. They also mention using highly desirable and expensive items as drains. This does not always work though, since it motivates players to earn more money.

Bilir (2009) analyses the whole economy of RuneScape, including its money supply. She identifies six faucets and eight drains. In her opinion, there is excess money supply in RuneScape and makes five suggestions for absorbing it. I myself did not find compelling evidence for the existence of excess money supply in the game, but she looks at a different time period, so this is not necessarily a contradiction.

Castronova (2001) has calculated a basic price index in the world of Norrath. He takes price data from a fansite and, since he cannot identify the standard bundle of goods, he gives each item an equal weight. His index shows a 29% deflation in one year. He puts the items in

two categories and finds that the items introduced in the new expansion lost 59% of their value, whereas the old items lost 17%.

In my opinion, Castronova's paper has captured the phenomenon of mudflation. Mudflation is coined from the words 'MUD' (Multi-User Dungeon) and inflation. It describes the price drops of old items which are rendered inferior by new ones. Expansions and updates are integral to MMORPGs. These games are frequently extended so that even old players can find new adventures. The expansions bring in new items that are better than the old ones in some respect, otherwise players would be dissatisfied. Initially, the supply of these new goods is zero, whereas demand is high. This results in elevated prices which decrease as more and more players obtain the item. When people finally buy the new item, the old one becomes useless, so they sell it. According to this logic, the prices of items should show deflation. First, their prices stabilise after release and they are rendered useless with newer updates.

3.2. THE EFFECT OF REAL MONEY TRADING

Opinions differ on whether real money trading causes inflation or not. Castronova (2006) argues that whenever players decide to sell their items once they quit, they plug an important drain and consequently accelerate inflation.

Heeks (2008) points out that gold farmers do not generally acquire game gold directly. Instead, they produce game items and sell those for game money. This way they do not actually increase the money supply, they increase the item supply.

I would like to argue that these two opinions are not actually in contrast. Castronova is talking about individual players engaging in real money trading once in a while, whereas Heeks is talking about companies mass-producing game gold. We can conclude that the separate categories of real money trading affect the price level differently.

3.3. THE MODEL

What we need to realise is that previous papers are all talking about the same model, they just focus on different aspects of it. They describe different phenomena that affect either the money supply or the demand for money, both of which change the equilibrium price level.

Yamaguchi (2004) mentions that consumption and saving choices have an effect on the demand for money. If economic agents are not motivated to save up and want to consume in the present, the demand for items increases. Players need money to purchase items, so this increases their demand for gold pieces, too. Heeks (2008) also examines the demand for money. If macros mass-produce items, the real income increases, as these constitute more goods to buy,

thereby creating a money shortage which causes deflation. The phenomenon of mudflation, as shown by Castronova's index, affects the demand for money in like manner. Similarly, adding new items to the economy means that real income increases and prices adjust by falling.

The money supply is not controlled by a central bank, it is the player community who determine it in the model of faucets and drains. Breau (2002), Lewis (2003), Füleki et al. (2008) and Bilir (2009) all use this model to describe the money supply. If more money comes through the drains than what leaves through the faucets, the money supply expands. In this case, $\lim_{t \rightarrow \infty} MS_t = \infty$. If the two amounts are exactly the same, the money supply stays constant. If less money comes in than leaves, then $\lim_{t \rightarrow \infty} MS_t = 0$.

It is nearly impossible to build a game where the faucets and drains are balanced, so the money supply will be converging to either zero or infinity. Of course, in a game with a growing player base, a money supply converging to zero is not a viable alternative, so the money supply of RuneScape is more likely to be converging to infinity. This is not necessarily a problem. When this convergence is very slow, players barely feel the effects. Besides, the creators always have the option to change the existing drains and faucets or add new ones if necessary.

Castronova (2006) highlights the fact that real money trading also has an effect on the money supply. By plugging a drain, it increases the amount of gold pieces in circulation.

In practice, all these phenomena can affect the price level at the same time and one cannot predict which is the strongest in advance. Moreover, there is no reason to assume that these factors are constant in time. Empirical work is thus needed to determine the changes in the price level.

4. RESEARCH FOCUS

I have three main questions I want to answer, all of which are connected to the price level of RuneScape.

First of all, I want to determine whether the faucets and drains of RuneScape are balanced. Instability can manifest itself either in deflation or inflation. Inflation is more likely though, as a growing player base means increasing demand for money. I am not saying that there cannot be temporary dips in memberships due to mass migrations away from the game, but as long as this is the exception rather than the general trend, it is irrational to let the money supply converge to zero, so hyperinflation can be an indicator of imbalanced faucets and drains.

Secondly, I want to find out whether real money trading causes inflation or deflation. This can be done, since in the game's life there have been bot-free and bot-infested periods. All I have to do is compare them and see whether the price level is significantly higher in the former or the latter case.

Finally, I want to find evidence of mudflation. RuneScape receives weekly updates, some of which bring new items. If these indeed make old ones worthless, that should appear as deflation. My Lasperyes-type consumer price index works with a fixed basket. The goods in it are the same throughout the entire period, which means they already existed on 20 May 2008, the day my index starts. I study a little more than four years, as the observed period ends on 26 June 2012. If mudflation is indeed a decisive factor, four years should be enough to see its effects.

To examine these phenomena, I create a consumer price index. I do not intend to copy real life price indices as they are meant to model a different reality. Instead, I consider the game's unique specialties and try to find the statistical tools that are the most suitable for depicting them.

First of all, there are less than 5000 types of tradable goods in the game and many of them are too rare or useless to be considered a typical consumption choice. The commonly used items can be put in a few small categories. Consequently, there is no need for subindices, a few representative items are enough to model reality.

Another thing worth considering is that the prices in this virtual economy are flexible. It would not occur to anybody to build a model on daily data with real life's sticky prices. It would mean a tremendous amount of extra work, but practically no additional information. In contrast, RuneScape's daily prices are easily accessible and hold information about the speed of certain

changes. I can thus calculate a fixed-base Lasperyes and a fixed-base Paasche index based on daily prices.

When looking at the results, we have to remember that the observed phenomena can have opposite effects. Imbalances in the faucet-drain model of money supply can cause inflation, whereas mudflation can cause prices to deflate. This might make drawing conclusions harder. For example, finding a constant price level can suggest that all factors are balanced or that their effects cancel each other out. What we can do is look at breakpoints in the index and check which factors were modified around that time.

5. ITEMS

In this chapter I show how my index is constructed. My first task is to find a basket that represents RuneScape's consumption choices. For this, I need to know the uses different items have and determine which of them can be considered consumption.

5.1. GAME ACTIVITIES

Most items have a specific purpose. They are needed for training a skill, finishing a quest or defeating a beast. Other items are just fun, but are not necessary for anything. Fortunately, these are not consumed in large quantities and therefore do not appear in my index. The ones that do, however, need to be categorised based on their function, the activity in which they are used. In order to do that, I dedicate this subchapter to introducing the most important in-game vocations and their effect on item demand.

What makes MMORPGs fun is the wide variety of activities available to its players. Adventurers have the opportunity to go on quests, battle monsters or other gamers, participate in minigames, train up to 25 skills and even to amuse oneself with distractions and diversions, events which break the tedium of skilling. However, only some of these activities generate demand for virtual goods.

Clearly, one needs herbs to train the herblore skill, logs to train firemaking and metal bars to train smithing. Players need these items in large quantities, because only a significant amount of practice makes perfect in this game. Other skills are meant for producing items rather than using them up. Fishermen can catch thousands of fish with their indestructible nets, woodcutters never need to change or even sharpen their axes and thieves do not need anything but their bare hands.

Skilling is not the only activity to consume items, so does combat. Fighters need potions to boost their combat skills, summoning familiars to aid them and food to heal their wounds. There are three styles of combat available: melee, ranged and magic. None of them is ultimately superior to others, everything depends on the opponent's combat style. This way players are encouraged to train all combat skills. Ranged and magic add further consumed items to the index: ammunition, such as arrows and runes.

Even though most armour and weapon types do not degrade and can be resold after use, I still add them to the index. The reason why I do this is that players tend to buy the best armour they can and keep it until they can afford a better one. They spend a considerable amount of

time saving up for the item and after purchase, their money is tied-up in it. Armour prices thus influence the amount of money players can spend on other goods and affect wellbeing.

Minigames, quests and distractions do not use up goods as a rule. Some items are provided during the activity, some can be resold after use and many are untradable to begin with. Therefore I can disregard such activities in the creation of my index.

Having examined RuneScape avocations, I found 12 skills 3 of which are combat styles that generate massive demand for game goods. Putting items in these 12 categories is suitable for representing the consumption choices of players.

5.2. CONSUMPTION

In this subchapter, I would like to explain how the aforementioned activities use up items. This is necessary for defining consumption, which is central to creating a consumer price index.

There are cases which resemble real life. Consuming an in-game pizza heals the player in combat. Even though its life-preserving effect saves people from monsters and not from starvation, pizza unquestionably belongs to the group of consumer goods.

Other items, such as seeds are more problematic. In real life, we plant seeds because we want to enjoy the beauty of flowers, the shade of large trees or the fruit they yield. The seed is just the raw material that has the potential to become something useful. However, RuneScape is completely different from real life in this aspect. Farmers buy expensive magic tree seeds and plant them only to end up with a tree that is not any different from the other magic trees in the realm. This process is not production, it is consumption. Farmers do not plant the seed in order to have a grown tree. All they want is to gain farming experience, because this increases their utility. The tree is just a by-product.

The most deceiving are the skills that produce a useful, tradable item. For example, herb-
lorists use herbs and secondary ingredients to create magic potions which can boost skills, cure poison or restore running energy. Players are willing to pay a large amount of money for these potions, so one might argue that herbs being the input of production should appear in a producers' price index and not a consumer price index. What this argument fails to consider is that creating potions is not profitable, the inputs cost more than the output. Despite this, players are still willing to make potions and sacrifice money. I should point out that we are talking about large amounts of money. Reaching level 99 herblore can cost up to 250 million gold coins. To put this in perspective, the best armour piece in the game, the Torva platebody only costs 212 million at the end of the examined period. So why are herb-
lorists paying all this extra cost when

they could just buy the potions? Basic microeconomics teaches us that companies cannot produce loss in the long run and yet herblore has never been profitable.

It seems that herblorists are also willing to pay for experience, just like farmers do. For them, creating potions means consuming experience. The price of this experience influences their well-being, so it needs to be represented in the CPI. If I put the herbs and secondary ingredients in the index, I represent the consumption choices of not only the potion user but the herblorist as well, hence the division of input prices into two parts: the output price and the difference between the output and input prices. This is a mathematical tautology: $p_{input} = p_{output} + (p_{input} - p_{output})$. The buyer pays p_{output} , the seller pays $p_{input} - p_{output}$ and together they pay p_{input} . My aim is to represent the consumption choices of the whole community, therefore I find this solution more adequate than using only the output prices.

Consumption and investment are inseparable in RuneScape. We do not know whether a player bought a weapon because he wants to make money fighting monsters or if he just wants to have fun challenging his friends to a duel. The tendency is that a player buys the weapon and uses it for both profit-oriented and fun-motivated combat. Therefore the weapon is partly a consumption item, its price changes influence the economic agent's well-being and it should be represented in a consumption price index.

Now that we have learned that in-game consumption can differ a great deal from what we experience in real life, I can move on to explaining a few other considerations about chosen items.

5.3. CHOSEN ITEMS

As players improve, their characters they gain access to better items. I find that medium-level items are the most suitable for representing the whole community. Training with low-level items is too slow and using high-level ones is unreasonably expensive, therefore neither can be considered typical.

I contemplate each skill's unique specialties when choosing representative items for them. Just to name one example, more skilled herblorists can create better potions and earn more experience than beginners. In contrast, using up an oak plank always yields the same construction experience, regardless of level. Consequently, for skills like herblore, whose optimal training method keeps changing, it is best to use more representative items than for stable skills like construction.

Whenever possible, I try to use items that are usually in RuneScape's top 100 most traded items list, because it represents typical consumption choices. However, expensive items are

never among the 100 most traded, so in their case I rely on my experiences. The chosen items can be found in *Table 1*.

Table 1: Chosen items

Melee	Magic	Ranged	Farming	Herblore	Crafting
Lobster	Air rune	Cannonball	Watermelon seed	Vial of water	Flax
Shark	Nature rune	Red chin-chompa	Irit seed	Snape grass	Blue dragon leather
Abyssal whip	Death rune	Adamant arrow	Limpwurt seed	Clean avantoe	Gold bar
Bandos chestplate	Astral rune	Adamant bolts	Pineapple seed	Mort myre fungus	Molten glass
Bandos tassets	Law rune	Karil's coif	Yew seed	Ranarr weed	
Amulet of fury	Cosmic rune	Karil's top		Clean toad-flax	
Ring of wealth	Soul rune	Karil's skirt		Crushed nest	
Dragon boots	Ahrim's hood				
Obsidian cape	Ahrim's robe top				
Dragon sq shield	Ahrim's robe skirt				
Slayer mask					
Smithing	Summoning	Fletching	Firemaking	Prayer	Construction
Gold ore	Swamp lizard	Yew logs	Maple logs	Dragon bones	Oak plank
Steel bar	Gold ring	Bow string	Magic logs	Big bones	
Adamant bar	Graahk fur	Yew long-bow (u)			
Iron bar	Raw chicken				

Source: edited by the author

6. EVENTS TO CONSIDER

To be able to determine how the standard bundle of goods changes over the time, which is essential for the Paasche index, I need to look at potential modifying events. This is necessary, because I do not have access to historical quantity data and I need to infer from game updates.

6.1. EVOLUTION OF COMBAT

Even though this is the most recent event under consideration, I discuss it first due to the magnitude of its effects. The arrival of 20 November 2012 brought fundamental changes in the combat system that affect each item related to fighting. Weapons lost their special attacks, armour gained the ability to increase lifepoints, the healing effects of food were altered and potion boosts were modified. As a result, a pre-EoC prayer potion is not the same item as a post-EoC prayer potion. These changes are so substantial and impact so many items that I cannot negate them using corrections.

Even though the Evolution of Combat was only released on 20 November 2012, economic agents were able to foresee its effects due to the preparations prior to the update. When players learned that certain items they possess were about to become useless, they were faced with the choice of still enjoying them until the update comes or immediately sell them to avoid serious losses. The choice to act immediately can impact the price level and its effects should not be included in the index. The mechanics of the new system were revealed on 26 June 2012, the day the Evolution of Combat beta was released. Therefore my index stops on that day.

6.2. BOT-FREE AND BOT-INFESTED TIMES

Like most MMORPGs, RuneScape also struggles with the presence of macros in the game. According to Castronova (2006), real money trading has costs for both players and game developers. Players have to endure inflation, overcrowded resources and spam. Developers could be faced with increased service provision costs.

Jagex also seems to recognise these costs. Besides having rules against macroing and real-world trading, they have also taken several anti-bot measures. On 2 January 2008, they removed free trade from the game. This meant that people could not pay significantly more or less for an item than its market price, they had to stay within a $\pm 5\%$ range. With this update, Jagex has successfully removed the motivation behind real world trading, but has also lost many players. Three years later developers felt their bot-detection systems were sophisticated enough to bring back free trade and on 1 February 2011 a new era started. Despite Jagex's best efforts,

the macros reappeared too. The next attempt to remove them was Bot Nuke Day, which happened on 25 October 2011. However, its effects were temporary.

In essence, the constant fight against macros has led to different economic periods. When macros are ubiquitous they produce resources in high quantities, driving their price down. In contrast, bot-free times mean higher prices and hence more expensive production skills. The use of third-party software seems to have a double-deflating effect: it decreases the prices of certain goods and make them relatively more common. The Paasche index is able to reflect both phenomena, whereas the Lasperyes can only capture the price drops due to its fixed basket.

In conclusion, the macro-related events in RuneScape history do not change the calculation the Lasperyes index, but they call for weight corrections on the Paasche index.

6.3. PLAYER-VERSUS-PLAYER COMBAT CHANGES

During the bot-free era of balanced trade, the player-killing system was also designed to make real money trading impossible. The drops of the victim were randomized, nobody could know for sure what they were going to receive if they were victorious. Due to this update, being killed on purpose could not be used as an alternate trading method.

Players argued that random PvP drops meant a new and easier way of obtaining special armour and weapons that normally required defeating a boss monster or winning a minigame. Jagex agreed that this indeed made the original source less rewarding and removed these drops from PvP worlds. The fallout was that certain goods became unreasonably rare. People could lose them on death, but the opponent could not receive them, hence they disappeared from the economy.

These changes caused boss-and minigame-only drop prices to skyrocket, while it is also reasonable to assume that some players substituted these items with cheaper ones. This is another reason why Paasche weights differ from Lasperyes ones.

6.4. BONUS XP WEEKENDS

Ever since 2010, every March and September RuneScape players get three special days when they can earn more experience than they normally would. On these days, resources are highly sought-after. What makes the event interesting is that it cannot be predicted when it will cause prices to skyrocket. The very first bonus experience weekend was the exception, since players were informed only two weeks before it happened. The following ones caused speculation about price rises and players hoping to earn a profit started hoarding items months before the event,

with the intention of selling them later. Sometimes prices only decreased once a Bonus XP Weekend was announced due to the mass supply provided by speculators.

Highly volatile prices could be smoothed out using a centred moving average, but this would lead to loss of relevant information in case of other changes. For example, the increase in price level caused by Bot Nuke Day was very sudden and since prices are flexible, my index can reflect that. In my opinion, the speed of the change is also relevant, so I abstain from smoothing prices.

7. THE LASPERYES INDEX

Having briefly introduced all major in-game events that affect the indices, I now move on to calculating the Lasperyes index. To do this, I need two types of data: prices and weights. I start with price data, because that is simple to download, then I explain the estimation of weights.

7.1. PRICES

As I mentioned in the Items chapter, I use goods in 12 categories to represent the market basket. Thanks to the data available on the main website, finding their prices is relatively easy. Even though the database only shows data from the past 180 days, there are fan sites that have been collecting data from the main website for years and go much further back. The data I downloaded from RuneScape Bits and Bytes date back to 20 May 2008. In the site's HTML code, prices are given in the following form: 1211328000: "1209". The 10-digit number is a time stamp marking the seconds elapsed since 1 January 1970. The second number in the quotation marks is the item price. My example shows that on 20 May 2008, 7:00:00pm EST, magic logs cost 1209 gold pieces.

7.2. LASPERYES WEIGHTS

The Lasperyes index works with a fixed basket, so I only need to find weights that more or less represent the consumed quantities and I can use them for the entire period. Unfortunately, there is practically no data available on consumed quantities. We only have the Top 100 most traded items list, but this is only available for the last 180 days and does not include all items in the index. Even fans do not seem to be interested in quantities, as there are no fansites that offer historical data on this list. The lack of data means that I need to calculate using estimated weights.

As I discussed before, the items in my index are either connected to skills or combat, since other activities do not generate mass demand for goods. Fortunately, skills have a toplist called Hiscores. From the Hiscores data, I can estimate how much experience players have in each skill and how many representative items are necessary to achieve that experience. The quantity of items used can serve as weights.

Regrettably, there are over a million players in the Hiscores, but one page only holds 22 names. Under these circumstances, downloading all data for 12 separate skills would take days. Instead, I acquire the necessary information by taking a sample. I generate a random number between 1 and 5000 for the first position, so that each item has a chance of getting in the sample

and then I take steps of 5000 names. This sample gives more accurate results than simple random sampling without replacement, because the list is ordered. Drawing only small or high numbers is not possible.

Since I find the current RuneScape website less suited for my purposes, I do not directly sample the Hiscores. Instead, I use a fansite, IndecentCode. Here it is possible to jump to any chosen page.

From the Hiscores sample I can calculate consumed items. I divide each skill into training phases. To give readers a sense of this process, beginner craftsmen spin bowstrings, from level 25 onwards, they make gold bracelets, at level 33 they switch to blowing glass into vials and at level 66 they become skilled enough to make blue dragonhide armour. I assume that all economic agents train using these methods and even level 99 players started with spinning bowstrings.

We do not know the exact time the players gained the experience we currently see in the Hiscores. This can cause distortions, since new players only experienced the cheap items of bot-infested times, whereas veterans might have lived and trained during the period of balanced trade. Fortunately, there have only been a few updates that made a skill significantly cheaper or more expensive. Generally it is true that relatively harder skills have always been relatively harder. Moreover, bots tend to make all observed skills easier, which mitigates the relative changes caused by them. While these weights are not perfect, using them gives more accurate results than assigning equal weights to all items, which is my only alternative.

What my quantities in absolute value show us is how many items economic agents consume during their in-game life, we just do not know how long that life is. This helps me determine the weights of the combat items. It is reasonable to assume that, in a world where weapons and armour cannot be broken beyond repair, a player only needs one piece of each armour and weapon type. Naturally, a player can lose an item in PvP combat, but in the current system the opponent gets the drop, so the average weapon/player ratio remains unaltered. In the Items chapter I mentioned that drops were randomized in the era of balanced trade and certain items were excluded from the drop list. Nevertheless, the players still need a new weapon if they lose the old one, they are just more likely to buy a lower level one. This will result in a change of the basket and will affect the Paasche index, but not the Lasperyes one. Therefore the Lasperyes weapon and armour weights can remain one.

7.3. RESULTS

Having collected the item prices in a matrix and the weights in a vector, I only need to calculate the result. I use a ‘for loop’ in MATLAB to do this. The code can be seen in the appendix. I entered the results into Excel and illustrated them in *Figure 1*.

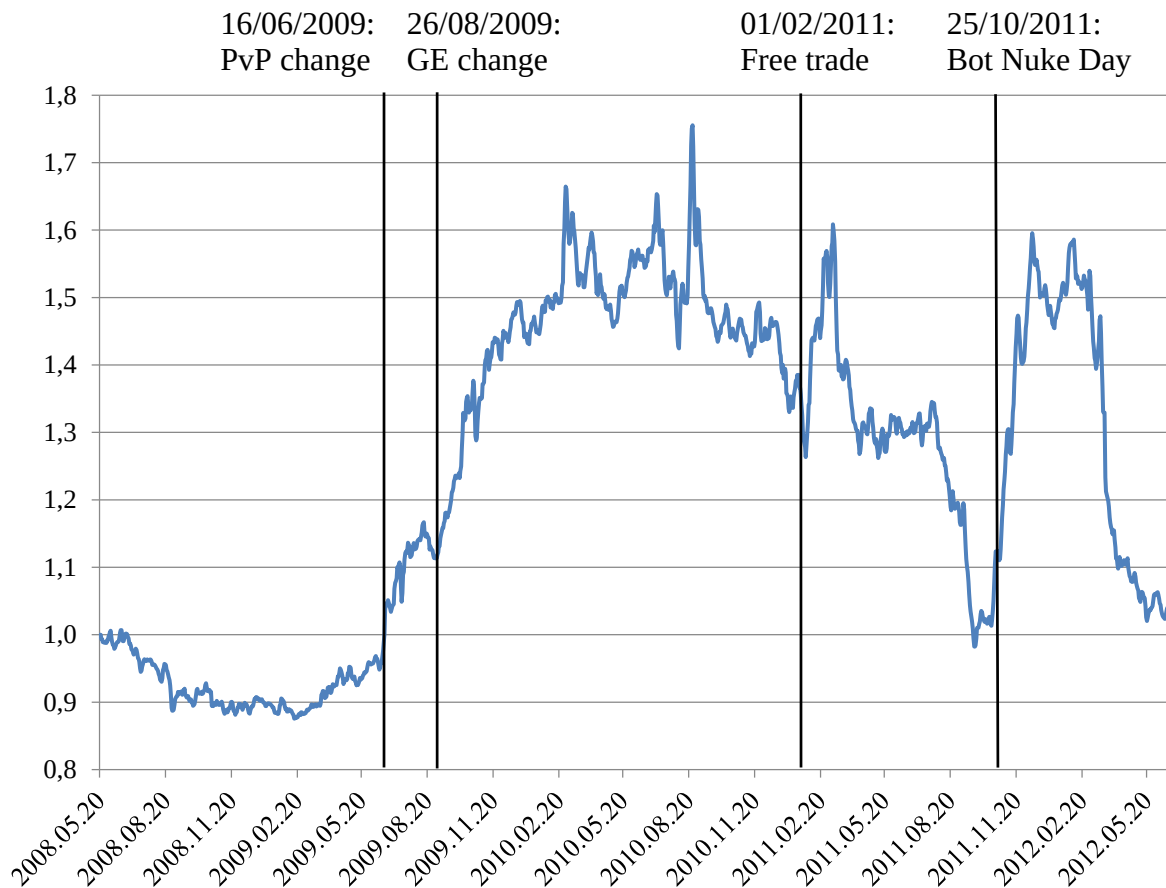


Figure 1: The Lasperyes Index

Source: drawn by the author

We can see that the prices are highly volatile, which can hide information. In the research focus chapter I mentioned that the key to drawing conclusions is looking at the breakpoints. Some of the bumps can be attributed to speculations regarding Bonus XP Weekends, but we can also identify four major events that caused more permanent changes.

The first was the exclusion of boss- and minigame-only items from the PvP drop table. This meant that if these weapons or armour pieces were lost in combat, they forever disappeared from the economy. On the one hand, this shrunk the supply, but on the other, it increased the demand, too, since the economic agents who lost these goods needed a replacement. Unsurprisingly, this caused armour and weapon prices to skyrocket.

What also increased the inflation was a Grand Exchange change. Before 26 August 2009 buying large quantities of items was serious work, because players could only buy the goods themselves and not a stackable bank note for a large amount of articles. Players can only hold up to 28 items in their inventory. Therefore if a player bought 1000 herbs they had to run to the bank 36 times and deposit them. This added extra costs to trading on the Grand Exchange and it was less tedious to buy directly from another player, who could offer a bank note for 1000 herbs. In contrast, selling 1000 items in the Grand Exchange did not require extra work. All the system recognised was that there was more supply than demand and that prices fell. Thus we have to be careful not to read too much into this deflationary period.

The return of free trade greatly influenced the price level as well. Many old players returned and everyone wanted to enjoy the old PvP system once again. This brought on a huge demand for items. Speculators appeared too, driving the prices further up. Eventually the enthusiasm died down, bots reappeared and deflation started. It did not last long, though.

Within a year, Jagex banned 98% of the bots and thereby demolished the cheap supply of resources. Prices skyrocketed again, only to plunge with the return of macros.

Before drawing the final conclusions, let us look at the Paasche index, too. This is calculated in the following chapter.

8. THE PAASCHE INDEX

Due to its fixed basket, the Lasperyes-type price index tends to exaggerate inflation. It does not consider the possibility of substituting expensive goods with cheaper ones. For this reason, I find it necessary to estimate a Paasche index in addition to the Lasperyes index. The price data is the same, so all I need to do is reweight it.

8.1. PAASCHE WEIGHTS

When the relative quantities of consumed items are unchanged, the Paasche weights are the same as the Lasperyes ones. There are two main differences. The first is the relative change in resource quantities due to the banning and return of bots. The second is putting lower-level weapon and armour pieces in the index when the boss-only ones became rare due to PvP changes.

Macros are the main source of raw materials in the game, since acquiring these items tends to be repetitive and can be easily automated. When macros are banned, the resource supply drops and prices skyrocket. Players of course take over collecting the items, but this job is tedious and the time players are willing to spend working is influenced by their utility functions.

At RuneFest 2011 in the ‘Breaking the bots’ insider session, Mark Michael Gerhard, the CEO of Jagex stated that after Bot-Nuking day, player activity decreased by 60%. This means that the maximum drop in the resource supply was 60%. Since I cannot estimate the exact drop in the supply, I am going to calculate the index for this extreme case. The price rises are given and the lower weights I assign to them, the lower the inflation gets (the prices of weapons did not skyrocket after Bot Nuke Day). Therefore I am calculating a best-case scenario.

I assume that the Paasche weights of resources in bot-free times are 40% of the Lasperyes ones and in bot-infested times they are the same. Also, these changes happened gradually. Even though macros were banned instantaneously on Bot Nuke Day, feeling the supply changes and getting to a new equilibrium took a while. The return of free trade was an even slower process. The petition started on 21 December 2010, the decision was made on 17 January 2011 and the changes happened on 1 February 2011. This means macro creators had more than a month to prepare. For these reasons, my weight corrections are also gradual. I gave a 2-month transition period for the return of free trade and 1 month for Bot Nuke Day.

Since MATLAB is more suited for handling large matrixes than Excel, I use it for modifying the weights. I put the weight corrections in a separate file. All of these corrections have

an absolute value between 0 and 1 and can be interpreted as percentages. I import the data into MATLAB and multiply each weights with $1 + \text{correction}$.

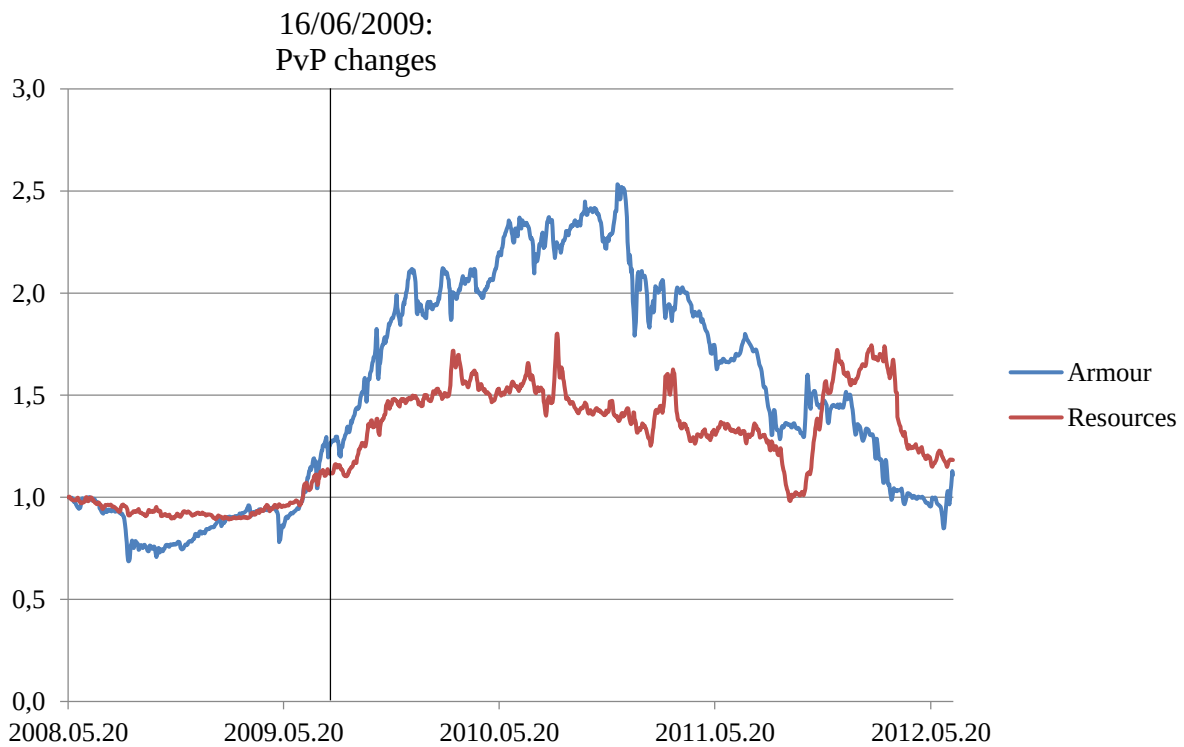


Figure 2: The effect of PvP drop changes

Source: drawn by the author

In case of the PvP changes, I also calculate the best case scenario. *Figure 2* indicates how armour and resource prices changed with the PvP update. Armour prices rose much faster than resource prices. Thus it is a rational choice to substitute expensive armour with a cheaper version and spend relatively more money on training skills. Due to lack of item quantity data, I do not know how many players made this choice. This is why I calculate the best case scenario in which all players switched. Since these changes were also gradual, I have a one-month transition period after the PvP changes and 2 months when they were removed by the return of free trade. These changes appear in the original quantity matrix.

8.2. RESULTS

My final results are illustrated in *Figure 3*. We can see that, as in real life, the Paasche index shows lower inflation than the Lasperyes index. We have to keep in mind, though, that the Lasperyes index is an actual estimation while the Paasche corrections show a best-case scenario. This is why a difference as high as 10 percentage points can be observed at times.

The reason why I calculated both indices is that I wanted to see if the fixed basket of the Lasperyes index causes any major distortions. Looking at both indices we can conclude that the general tendencies are the same. The same four changes can be observed that were already visible in the Lasperyes index. Knowing this, I can now move on to drawing conclusions.

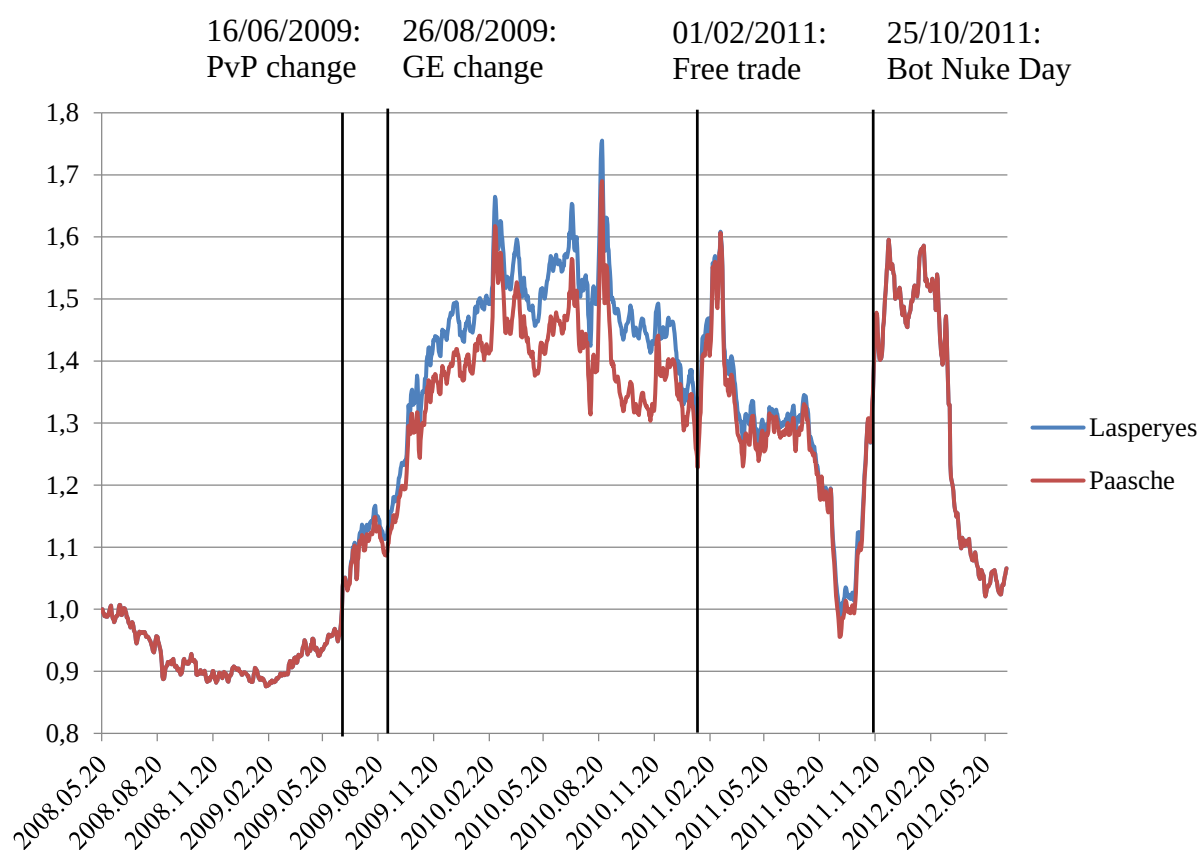


Figure 3: The results

Source: drawn by the author

9. CONCLUSION

I attempted to answer three inflation-related questions by creating a consumer price index. I wanted to know if imbalanced faucets and drains are causing inflation, if macros affect the price level and if the game is faced with substantial mudflation. Unfortunately my results do not give all the answers, although I did find something I was not even looking for.

Based on this outcome I cannot say with absolute certainty if there is excess money in the game that would cause inflation. All I can say for sure is that it is not causing hyperinflation. The rest of the information is hidden by the strong influence of the demand for money and the volatility of the index. However, this can also be considered a result. Previous papers mostly focus on the money supply. I have demonstrated that the demand for money can be even more important and deserves more attention. Its influence on the RuneScape price level is obvious, whereas the inflating effects of the money supply, assuming there are any, are negligible. I am not saying that money supply is irrelevant or unable to impact the price level, merely that in this particular game the faucet-drain system works well enough not to cause significant inflation. Kudos, Jagex.

My second focus was on the activities of macros. We can see that the price level is much lower in periods when macros were out and about. This lets us conclude that the comparative advantage of third-party software lies in collecting items and not in directly creating money.

The last thing I was able to determine is that mudflation does not significantly influence the overall price level. If it did, then the Lasperyes index would show constant deflation. Nevertheless, mudflation might affect individual items, but definitely not the economy as a whole.

WORKS CITED

Literature

- Bilir, T. E. (2009): *Real Economics in Virtual Worlds: A Massively Multiplayer Online Game Case Study: Runescape*. <http://ssrn.com/abstract=1655084> (Accessed: March 14, 2013.)
- Breaking the Bots Insider Session: <http://www.youtube.com/watch?v=VZHalHeWer8> (Accessed: March 12, 2013.)
- Breau, G. (2002): *Online World Economy*. <http://www.quantumg.net/owe.php> (Accessed: March 14, 2013.)
- Castronova, E. (2001): *Virtual Worlds: A First-Hand Account of Market and Society on the Cyberian Frontier*. CESifo Working Paper Series No. 618. <http://ssrn.com/abstract=294828> (Accessed: March 14, 2013.)
- Castronova, E. (2006): A cost-benefit analysis of real-money trade in the products of synthetic economies. *Info*, Vol. 8, Issue 6, pp. 51–68.
- Füleki, D. – Theiss-Balázs, Z. – Balkányi, P. – Pocsarovszky, K. (2008): A magyar webes piac technológiai architektúrái (Technological Architectures Of The Hungarian Web Market). *Vezetéstudomány*, Vol. 39, Issue 12, pp. 39–50.
- Heeks, R. (2008): *Current Analysis and Future Research Agenda on "Gold Farming": Real-World Production in Developing Countries for the Virtual Economies of Online Games*. http://www.sed.manchester.ac.uk/idpm/research/publications/wp/di/di_wp32.htm. (Accessed: March 14, 2013.)
- Lewis, S. (2003): *Money Supply Impacts on an Online Economy*. White Paper, http://www.flyingscythemonkey.com/Money_Supply_White_Paper.htm (Accessed: March 14, 2013.)
- Yamaguchi, H. (2004): *An Analysis of Virtual Currencies in Online Games*. Komazawa University <http://dx.doi.org/10.2139/ssrn.544422>. (Accessed: March 14, 2013.)

Databases

- Indecent Act's Unofficial RuneScape HiScores: <http://www.indecentcode.com/hs/index.php?id=S%20U%20%20O%20%20M%20I> (Accessed: February 2, 2013.)
- RuneScape Bits and Bytes. <http://www.rsbandb.com/runescape-grand-exchange-portfolio/> (Accessed: February 1, 2013.)
- RuneScape Grand Exchange Database: http://services.runescape.com/m=itemdb_rs/frontpage.ws (Accessed: March 15, 2013)

APPENDIX

The MATLAB code used for calculating indices:

```
function tdkindexek
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Lasperyes %%%%%%%%%
q=xlsread('laTDKqvektor');
p=xlsread('laTDKarmatrix');

[m, n]=size(p);

q0p1=zeros(m, 1);

for i = 1:m
    for j= 1:n
        q0p1(i, 1)=q0p1(i, 1)+q(1, j)*p(i, j);
    end
end

xlswrite('matlabindex.xls', q0p1, 'q0p1');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Lasperyes %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Paasche %%%%%%%%%
q=xlsread('paTDKqmatrix');
p=xlsread('paTDKarmatrix');
qk=xlsread('paTDKqkorrmatrix');

[m, n]=size(p);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% súlyvektor korrekció
q=q.*(1+qk);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% súlyvektor korrekció
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% q1p1/q1p0
q1p0=zeros(m, 1);
for i = 1:m-1
    for j= 1:n
        q1p0(i, 1)=q1p0(i, 1)+q(i+1, j)*p(1, j);
    end
end

q1p1=zeros(m, 1);

for i = 1:m-1
    for j= 1:n
        q1p1(i, 1)=q1p1(i, 1)+q(i+1, j)*p(i+1, j);
    end
end

xlswrite('matlabindex.xls', q1p0, 'q1p0');
xlswrite('matlabindex.xls', q1p1, 'q1p1');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Paasche %%%%%%%%%
end
```