

**ТЕХНИЧЕСКИ УНИВЕРСИТЕТ - СОФИЯ**

ДИСЦИПЛИНА

„СИСТЕМНО ПРОГРАМИРАНЕ”

Летен семестър 2015/2016г.

Задача 9

Изработили:

Благой Николов Натали Христова

Фак.номер: 121213206 и 121213042  
Група 43

**Увод**

**Условие**

Задача 9. В супермаркетите и повечето себеуважаващи се чалготеки всеки продукт има баркод, чрез който бива идентифициран. С цел да помогнете на управител на подобно учреждение реализирайте приложение, което приема даден набор от прочетени баркодове и накрая извежда сумата, която трябва да бъде заплатена. Използвайте минимум два процеса. Имайте предвид, че няколко каси могат да работят едновременно!

Допълнително: Дайте възможност да автоматично изчисляване на ресто при подадена сума от клиента.

Забележка: Използвайте минимум 10 артикула.

**Задачи**

Една от най-главните задачи, която трябва да се изпълни, е реализацията на многонишков клиент-сървър. Трябва да има двустранна връзка, защото главната функционалност на програмата е следната: крайният потребител въвежда своя избор, клиента изпраща информацията към сървъра, сървърът я обработва (прави изчисления) и връща отговор. За изчислението на общата сума трябва да съберем цените на всички артикули, желани от клиента. Всеки артикул трябва да съдържа следните характеристики: баркод, по който да бъде разпознаван; име на продукта, защото клиентът ще иска да знае за какво плаща; единична цена на продукта. За допълнителното условие е необходимо от дадена въведена сума от потребителя (клиента) да бъде извадена общата сума на артикулите и остатъкът да се бъде изписан като ресто. Ако въведената сума от клиента е по-малка от общата сума, ще има съобщение, което ще уведоми клиента,

**Глава 1: Технологии и похвати**

За комуникацията между сървъра и клиента използваме сокет от тип: SOCK\_STREAM, който използва TCP протокол. Използваме perror за принитиране на грешките в stderr. Създаваме дъщерни процеси с fork(), където се извършват изчисленията (обработват се данните, изпратени от клиента), а бащиният процес чака за нови клиенти. За описването на продуктите използваме структура от данни Product, където всеки продукт има баркод, име и цена.

**Глава 2:** **Функции**

**Функции на многонишковия сървър:**

1. Главна функция: int main(int argc, char \*argv[]) на сървъра:   
   Създава се сокетът на сървъра със стандартната функция: sockfd = socket(AF\_INET, SOCK\_STREAM, 0). Инициализираме сокет структурата да е равна на нула с функцията: bzero((char \*)&serv\_addr, sizeof(serv\_addr)). Инициализираме номера на порта с portno = 5001. Байндваме хост адреса с функцията: (bind(sockfd, (struct sockaddr \*) &serv\_addr, sizeof(serv\_addr)). С функцията listen(sockfd, 5) слушаме за поява на нова връзка, като първият аргумент е дескрипторът на сокета, а вторият е размерът на опашката за неизпълнените задачи. Максималният размер разрешен за повечето системи е 5. В безкраен while цикъл инициализираме новия файлов дескриптор: newsockfd = accept(sockfd, (struct sockaddr \*) &cli\_addr, &clilen), по този начин клиентът се свързва към сървъра. В цикъла чрез fork() правим нов дъщерен процес за всеки клиент. При дъщерен процес се вика функцията void doprocessing(int sock), която отговаря за обработването на клиентските съобщения, както и за изчисленията на сумата на продуктите.
2. Функция за обработка на клиентското съобщение: void doprocessing(int sock): В нея сървърът чете получената стойност от клиента, конвертира я от network към host long с функцията: int product\_id = ntohl(received\_int). Ако полученото число е 0, прекъсваме цикъла чрез break. На променлива присвояваме извлечената цена на продуктите чрез функцията getProductPrice(product\_id) и сумираме цените. След това изпращаме през сокета резултата към клиента.
3. Функция за извличане на цена от продуктите: double getProductPrice(int barcode):  
   Тук описваме всеки един продукт с неговите характеристики (id, име и цена), след което ги вкарваме в масив. С for цикъл обхождаме масива и за всеки един елемент и проверяваме дали съвпадат barcode и id на продукта. Когато съвпадат, връщаме цената на продукта.

**Функции на клиента:**

1. Главна функция: int main(int argc, char \*argv[]) на клиента:   
   Извикваме функцията printMenu(), за да изведем на екрана всички продукти. Използваме portno = atoi(argv[2]), за да конвертираме стринга от числа към int.Създава се сокетът на клиента със стандартната функция: sockfd = socket(AF\_INET, SOCK\_STREAM, 0). С фукнцията server = gethostbyname(argv[1]), взимаме името като аргумент и връщаме указател към съответния хост. Свързваме се със сървъра чрез (connect(sockfd, (struct sockaddr\*)&serv\_addr, sizeof(serv\_addr)). Сега чрез безкраен цикъл for (;;) проверяваме стойността на числото, въведено от клиента. Ако числото е по-малко от 0 или по-голямо от 10 казваме, че няма такъв продукт. В противен случай директно пишем в сокета и изпращаме числото към сървъра за обработка чрез int converted\_number = htonl(number\_to\_send). Ако числото е 0 прекъсваме цикъла и изпращаме числото за обработка към сървъра. След това чрез read(sockfd, &sum, sizeof(sum)) получаваме отговора на сървъра с вече изчислената сума и я извеждаме на екрана. Накрая викаме функцията printChange за изчисляване на ресто.
2. Функция за изчисляване на ресто: void printChange(double sum):  
   Първо извеждаме съобщение за въвеждане на даден брой пари за плащане. След което изчисляваме на база сумата от сметката и въведените пари какво е рестото. Докато рестото е по-малко от 0, извеждаме съобщение потребителят да въведе нова сума. Накрая извеждаме на екрана стойността на полученото ресто.
3. Функция за принтиране на всички продукти: void printMenu():  
   Тук описваме всеки един продукт с неговите характеристики (id, име и цена), след което ги вкарваме в масив. С for цикъл обхождаме масива и принтираме продуктите.

**Глава 3: Указания**

1. За да тръгне приложението, трябва да стартираме клиента и сървъра.
2. От терминала компилираме и стартираме първо сървъра при подаден изпълним файл.
3. След това от нов терминал, компилираме и стартираме клиента при подаден изпълним файл, хост и порт. Хостът трябва да бъде localhost, а порта 5001.
4. При стартирането на клиента на екрана ще се изпишат продуктите, след което ще има съобщение за въвеждане на id на желания от клиента продукт/и. Много е важно да се използва терминираща 0 за край на въвеждането.
5. След това ще се появи статуса на желаното от потребителя пране.

**Глава 4: Тестови данни**

**Заключение**

Реализираната от нас програма отговаря напълно на поставените условия и задачи включени в заданието. Имплементираното клиент-сървър приложение ще помогне значително на управителите на себеуважаващите се учреждения да управляват сметките си. Също така е реализирана и допълнителната задача, която автоматично изчислява ресто при подадена сума от клиента. Доволни сме от себе си и си поставяме оценката Отличен (5,50).

**Приложение**

Линк към source файловете: <https://github.com/BlagoyNikolov/SystemProgramming>