**УНИВЕРЗИТЕТ "СВ. КЛИМЕНТ ОХРИДСКИ"**

Факултет за

Информатички и комуникациски технологии

**СТРУКТУРИРАНО ПРОГРАМИРАЊЕ**

Семинарска работа



Задача број 45

Кодирање и декодирање на текст според даден клуч

Студент: Иван Петковски Професор: Рамона Маркоска

Битола, 2022

# Дефинирање на проблемот

Да се напише програма во која со помош на две функции, кои ќе се одбираат преку мени, ќе се овозможи кодирање или декодирање на даден стринг со должина најмалку до 400 карактери. Во стрингот смее да има само букви, бројки и празни места.

Кодирањето да се врши според следното правило:

* Се внесува позитивен цел број при секое стартовање на програмата, кој да се обезбеди да не биде повеќе од 10 цифри. Понатаму да се пресмета бројот на неговите цифри. ASCII кодот на секој карактер да се наголемува за врендоста која го претставува бројот на цифри. (Пример: ако се внесе 2345, ASCII кодот на секој карактер при кодирањето да се наголеми за 4 цифри).
* При декодирање да се користи истиот број на начин кој ќе овозможи инверзно враќање на кодиран стринг во неговиот првобитен облик.
* Програмата да се тестира со три стринга, кои ќе бидат закодирани и истите тие, понатаму, во програмата, декодирани, со избор на функцијата за декодирање.

# 

# Опис на постапката

Самиот збор ASCII кодирање преставува премин од еден знаковен тип (char) во целоброен тип на податок(int) и обратно.

Главната задача на проблемот е кодирање и декодирање на внесен текст според внесен клуч. Стрингот кој се внесува, се внесува со помош на циклус со контрола кој може да содржи само букви бројки и празни места. (со помош на do/while се врши контрола за точно внесен стринг). Бројот т.е. клучот за кодирање се внесува со конторла бидејќи негов услов е да не е поголем од 10 цифри. Тоа се врши со низа од карактери кои се проверуваат дали сите внесени елементи се цифри и се определува бројот на цифри во внесен елемент.

Во главната после внесувањето на текстот и клучот за кодирање на екран се прикажува мени за избор.

Доколку корисникот избере 1, од менито за избор се врши повик на функцијата за кодирање која не враќа вредност а прима три влезни аргументи.

void funkcija1(char VnesenTekst[],int DolVnesenText,int brPomesti)

Доколку корисникот избере 2, од менито за избор се врши повик на функцијата за декодирање која не враќа вредност а прима три формални аргументи.

void funkcija2(char VnesenTekst[],int DolVnesenText,int brPomesti)

Доколку корисникот избере 3 од менито за избор се врши излез од програмата и крај за кодирање и декодирање.

# Листинг на кодот

#include<iostream>

#include<cstring>

using namespace std;

//definicija na funkcija za kodiranje

void funkcija1(char VnesenTekst[],int DolVnesenText,int brPomesti)

{

//deklaracija i inicijalizacija na promenlivi

int min1=65,min2=97,max1=90,max2=122;

for(int i=0;i<DolVnesenText;i++)

{

//odreduvanje na ascii kodovi

int ascibroj=(int)VnesenTekst[i];

//mali ascii

if(ascibroj>=min2 && ascibroj<=max2)

{

if(ascibroj+brPomesti<=max2)

VnesenTekst[i]=(char)((int)VnesenTekst[i]+brPomesti);

else

VnesenTekst[i]=(char)(min2+((int)VnesenTekst[i]+brPomesti)-1-max2);

}

//golemi ascii

if(ascibroj>=min1 && ascibroj<=max1)

{

if(ascibroj+brPomesti<=max1)

VnesenTekst[i]=(char)((int)VnesenTekst[i]+brPomesti);

else

VnesenTekst[i]=(char)(min1+((int)VnesenTekst[i]+brPomesti)-1-max1);

}

}

cout<<endl;

//pecatenje na kodiran string

for(int i=0;i<DolVnesenText;i++)

cout<<VnesenTekst[i];

cout<<endl;

}

//definicja na funkcija za dekodiranje

void funkcija2(char VnesenTekst[],int DolVnesenText,int brPomesti)

{

//deklaracija i inicijalizacija na promenlivite

int min1=65,min2=97,max1=90,max2=122;

for(int i=0;i<DolVnesenText;i++)

{

//odderuvanje na ascii kod

int ascibroj=(int)VnesenTekst[i];

//golemi bukvi

if((ascibroj>=min1 && ascibroj<=max1)||(ascibroj>=min2 && ascibroj<=max2))

{

//mali bukvi

if(ascibroj>=min2 && ascibroj<=max2)

{

if(ascibroj-brPomesti>=min2)

VnesenTekst[i]=(char)((int)VnesenTekst[i]-brPomesti);

else

VnesenTekst[i]=(char)(max2-(min2-((int)VnesenTekst[i]-brPomesti))+1);

}

//golemi bukvi

if(ascibroj>=min1 && ascibroj<=max1)

{

if(ascibroj-brPomesti>=min1)

VnesenTekst[i]=(char)((int)VnesenTekst[i]-brPomesti);

else

VnesenTekst[i]=(char)(max1-(min1-((int)VnesenTekst[i]-brPomesti))+1);

}

}

}

cout<<endl;

//pecatenje na dekodiran tekst

for(int i=0;i<DolVnesenText;i++)

cout<< VnesenTekst[i];

cout<<endl;

}

int main()

{

char VnesenTekst[400];

char brPomesti[20];

int izb,brbroj;

int DolVnesenText,cifra,brcifri=0,brPomestiac=0;

//ciklus za kontorla na vnesen tekst

do

{

cout<<"\n Vnesete tekst za obrabotka ->>> ";

cin.getline(VnesenTekst,400);

DolVnesenText=strlen(VnesenTekst);

brPomestiac=0;

for(int i=0;i<DolVnesenText;i++)

if(!isalpha(VnesenTekst[i]) && !isdigit(VnesenTekst[i]) && VnesenTekst[i]!=' ')

brPomestiac=brPomestiac+1;

if(brPomestiac!=0)

cout<<"\n GRESKA POVTORNO VNESETE !!!! ";

}while(brPomestiac!=0);

int pombrPomesti=0;

//ciklus za kontorla na vnesne broj

do

{

cout<<"\n Vnesete broj ->>> ";

cin.getline(brPomesti,20);

brbroj=strlen(brPomesti);

pombrPomesti=0;

for(int i=0;i<brbroj;i++)

if(isdigit(brPomesti[i]))

pombrPomesti++;

if(!(pombrPomesti==brbroj && brbroj<=10))

cout<<"\n GRESKA OBIDETE SE POBTORNO !!!! "<<endl;

}while(!(pombrPomesti==brbroj && brbroj<=10));

//meni za izbor

do

{

cout<<"\n 1. Kodiranje ";

cout<<"\n 2. Dekodiranje ";

cout<<"\n 3. Izlez ";

cout<<"\n ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^";

cout<<"\n Vnesete izbor : ";

cin>>izb;

cout<<"\n ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^";

switch(izb)

{

case 1: funkcija1(VnesenTekst,DolVnesenText,brbroj);

break;

case 2: funkcija2(VnesenTekst,DolVnesenText,brbroj);

break;

case 3: break;

default:cout<<"\n POGRESEN IZBOR "<<endl;

}

}while(izb!=3);

cout<<"\n KODIRANJETO I DEKODIRANJETO SE ZAVRSENI ---- PROGRAMATA ZAVRSI "<<endl;

return 0;

}

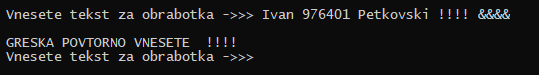
# Извршни екрани

Кога клиентот ќе ја стартова програмата на екран се прикажува порака за внесување на текст .



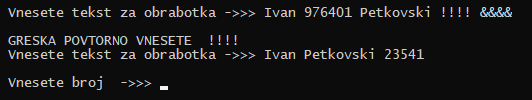
Слика 1.

Доколку се внеси стринг кој не е коректен на екран се прикажува порака за погрешен внес и одма во нов ред порака за внесување на нов стринг.



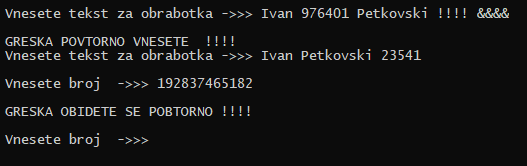
Слика 2.

После внесување на коректен стринг на екран се прикажува порака за внесување на број.



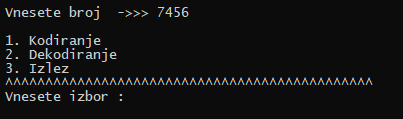
Слика 3.

Доколку корисникот внесе погрешен број кој не се соодветствува со условот од задачата на екран се прикажува порака за грешка и приказ на нова порака за следен внес.



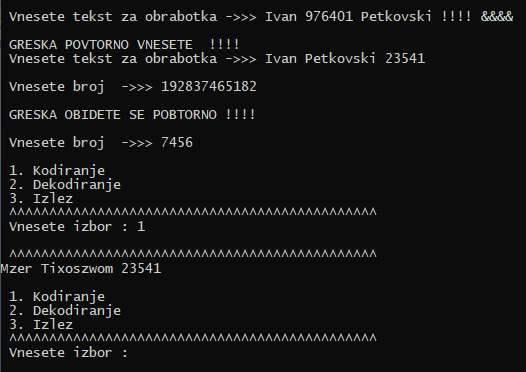
Слика 4.

По внесот за број кој ги исполнува условите од дефинираниот проблем на екран се прикажува мени со можност за избор од менито кое е прикажано на слика 5.



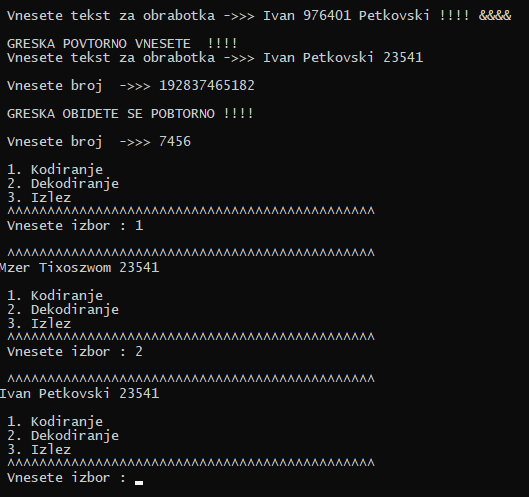
Слика 5.

Ако клиентот кој ја користи апликацијата избере од менито за избор 1, со што се повикува функцијата за кодирање на внесен стринг на екран се прикажува кодираниот стринг и повторно менито за избор со следна можност за избор.



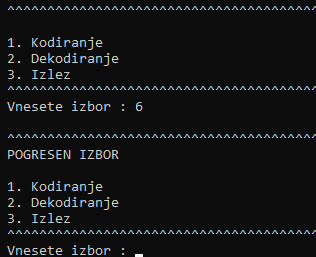
Слика 6.

На слика 7 се прикажува изборот од менито со број 2 за декодирање на внесниот текст. По изборот на еркан е прикажан декодираниот текст и можност за следен избор од менито.



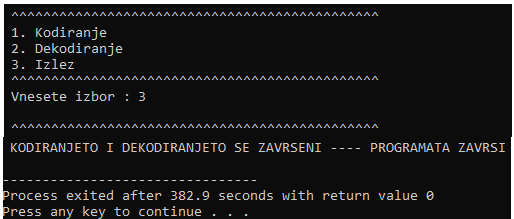
Слика 7.

Ако од менито се избере број кој не е понуден во можностите за избор на екран се прикажува порака за погрешен избор и повторно приказ на менито. (Слика 8)



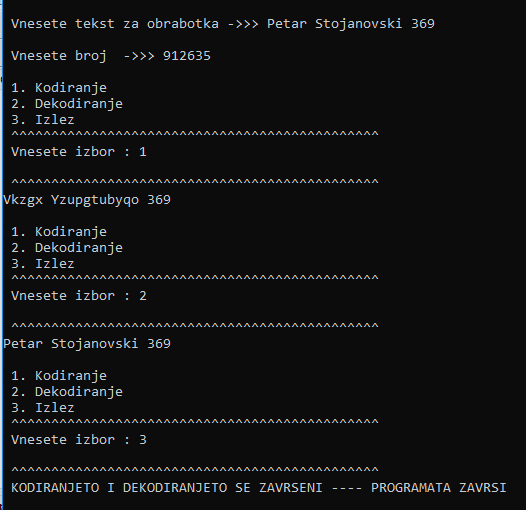
Слика 8.

Доколку корисникот има потреба од крај за приказ на менито и крај на сите можности потребно е да внесе број 3. По внесување на број 3 на екран се прикажува порака за крај на кодирање и декодирање.



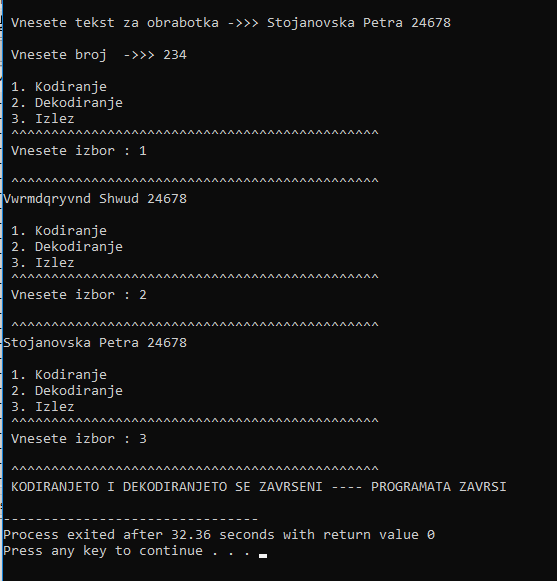
Слика 9.

ТЕСТ 2:



Слика 10

ТЕСТ 3



Слика 11