

ВТОР ПАРЦИЈАЛЕН ИСПИТ ПО

ПРОГРАМИРАЊЕ И АЛГОРИТМИ

18.1.2021

Низи :

Да се напише програма во која ќе се внесе должина на низа од тастатура, а потоа ќе се внесат и елементите на низата. Дополнително од тастатура се внесува и природен број h , кој треба да биде помал од должината на низата. Ако бројот h е поголем од доделената должина, програмата треба да отпечати порака за грешка и да заврши. Ако внесениот број h е помал или еднаков на доделената должина, тогаш во низата треба да се пронајдат сите поднизи од последователни елементи со должина h . За секоја од поднизите треба да се направи проверка дали го содржат бројот h . Програмата треба да ги отпечати оние поднизи кои го содржат тој број. Ако не постои таква подница, програмата појавува порака за грешка и завршува.

Пример:

niza = [4, 15, 1, 42, 52, 5, 39, 2, 18, 87, 91]

За внесен број $h = 5$, се разгледуваат поднизите: [4, 15, 1, 42, 52], [15, 1, 42, 52, 5], [1, 42, 52, 5, 39], [42, 52, 5, 39, 2], [52, 5, 39, 2, 18], [5, 39, 2, 18, 87], [39, 2, 18, 87, 91].

Програмата треба да ги испечати оние низи кои во случајот го содржат бројот 5. Тие низи се: [15, 1, 42, 52, 5], [1, 42, 52, 5, 39], [42, 52, 5, 39, 2], [52, 5, 39, 2, 18], [5, 39, 2, 18, 87].

Решение:

```
#include <stdio.h>
```

```
int main(){
    int niza[100],n,h,i,j,p,dali=0;

    printf("Vnesi ja dolzinata na nizata\n");
    scanf("%d",&n);
    if(n>100){printf("Dolzinata na nizata nadminuva maksimum");return 0;}
    for(i=0;i<n;i++){
        printf("niza[%d]=",i);
        scanf("%d",&niza[i]);
    }
    printf("Vnesi ja vrednosta za h\n");
    scanf("%d",&h);
    if(h>n){
        printf("Vnesenata vrednost za h e pogolema od dolzinata na nizata");return 0;
    }
    for(i=0;i<n-h;i++){
        // pocetok na podnizata
        p=0;
        for(j=i;j<i+h;j++){
            // izminuvanje na podnizata
            if(niza[j]==h){
```

```

    // najden e elementot
    p=1;
    break;
}
}
if(p==1){
    dali++;
for(j=i;j<i+h;j++){
    printf("%d\t",niza[j]);
}
printf("\n");
}
}
if(dali==0){
    printf("Nema nitu edna podniza sto go sodrzi brojot %d",h);
}
return 0;
}

```

Матрици:

Потребно е да се напише програма за работа со матрици. Програмата треба да дозволи внесување на димензиите на една квадратна матрица A, со целобројни елементи, со максимални димензии (100x100). Елементите на матрицата и димензиите на матрицата се читаат од тастатура. Од почетната матрица да се креира нова матрица B, со исти димензии како и почетната матрица. Матрицата B треба да се креира од матрицата A, така што ќе се ископираат елементите од главната и споредната дијагонала, и сите останати елементи освен елементите кои се десно од главната дијагонала а лево од споредната дијагонала. Елементите кои се десно од главната а лево од споредната дијагонала ќе се пополнат со вредност 0. На крај да се отпечати ново-формираната матрица B.

$A = \begin{bmatrix} 1 & 17 & 2 & 5 & 13 \\ 2 & 5 & -2 & -1 & 2 \\ 15 & 12 & 1 & 2 & 3 \\ 2 & -2 & -33 & 23 & 8 \\ 9 & 1 & 2 & 3 & 4 \end{bmatrix}$	$B = \begin{bmatrix} 1 & 0 & 0 & 0 & 13 \\ 2 & 5 & 0 & -1 & 2 \\ 15 & 12 & 1 & 2 & 3 \\ 2 & -2 & -33 & 23 & 8 \\ 9 & 1 & 2 & 3 & 4 \end{bmatrix}$
---	---

Решение:

```
#include <stdio.h>
```

```

int main(){
int A[100][100],n,i,j,B[100][100];
// за тестирање
// int C[100][100]={ {1,17,2,5,13},{2,5,-2,-1,2},{15,12,1,2,3},{2,-2,-33,23,8},{9,1,2,3,4}};
printf("vnesi ja dimenzijata na kvadratnata matrica A");
scanf("%d",&n);
if(n>100){printf("Dimenzijata na matricata go nadminuva maksimumot");return 0;}
for(i=0;i<n;i++){
    for(j=0;j<n;j++){

```

```

printf("A[%d][%d]=",i,j);
scanf("%d",&A[i][j]);
B[i][j]=0;
}
}
// prv nacin
for(i=0;i<n;i++){
    for(j=0;j<n;j++){
        if(!(j>i&& i+j<n-1)){
            B[i][j]=A[i][j];
        }
    }
}

// vtor nacin
/* for(i=0;i<n;i++){
    for(j=0;j<n;j++){
        if(j>i&& i+j<n-1){
            B[i][j]=0;
        }else
            {B[i][j]=A[i][j];}
    }
} */

for(i=0;i<n;i++){
    for(j=0;j<n;j++){
        printf("%d\t",B[i][j]);
    }
    printf("\n");
}
return 0;
}

```

Датотеки:

Да се напише програма која од командна линија добива имиња на **една влезна** и **една излезна** датотека. Програмата треба да ја измине влезната датотека и за секој ред да го најде бројот на големи букви и бројот на самогласки. Притоа, **во излезната датотека** да ги отпечати оние редови во кои има повеќе големи букви од самогласки. На крај **во излезната датотека** да се испечатат бројот на големи букви и бројот на самогласки кои се појавиле во целата датотека. Исто така, да се провери дали правилно се користи програмата, ако не, да се даде упатство за користење. Линиите во датотеките имаат најмногу 250 знаци.

Пример:

Влезна датотека:
PiA i PSP

Излезна датотека:
PiA i PSP

Matematika 1 i Matematika 2
OEL i OEK
Fizika 1 i Fizika 2

OEL i OEK
15 26

Решение:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
int main(int argc, char *argv[]){
FILE *fin,*fout;
int i;char c;
char niza[251];
int brgolemi,brsamoglaski,brgline,brsamline;
if(argc!=3){
printf("Nepравilen povik. %s datIN datOUT\n",argv[0]);return 0;
}
if((fin=fopen(argv[1],"r"))==NULL){
printf("Ne moze da se otvori vlezna datoteka\n");return 0;
}
if((fout=fopen(argv[2],"w"))==NULL){
printf("Ne moze da se otvori izleznata datoteka\n");return 0;
}
i=brgline=brgolemi=brsamline=brsamoglaski=0;

while((c=fgetc(fin))!=EOF){
if(c!='\n'){
niza[i]=c;
i++;
if(isupper(c)){brgline++;brgolemi++;}
c=toupper(c);
if(c=='A'||c=='E'||c=='I'||c=='O'||c=='U'){
brsamline++;brsamoglaski++;
}
}
}
else {

if(brgline>brsamline){
niza[i]=0;
// I nacin ne raboti so prazni mesta
// fprintf(fout,"%s\n",niza);
// II nacin treba da se pogrizime za nov red
// fputs(niza,fout);
// III nacin
i=0;
while(niza[i]){
```

```
    fputc(niza[i],fout);
    i++;
}
fputc(c,fout);

}
    brsamline=brgline=0;
    i=0;
}
}
fprintf(fout,"%d %d\n",brgolemi,brsamoglaski);
fclose(fin);
fclose(fout);
    return 0;
}
```