

ПРВ ПАРЦИЈАЛЕН ИСПИТ ПО
ПРОГРАМИРАЊЕ И АЛГОРИТМИ

Термин 1 Група 1 18.11.2019

1. Да се напише програма која ќе отпечати колку штета може да нанесе даден играч во некоја игра. Секој играч има одредено време (временски период) во кој може да нанесе удари. Секој играч има право да нанесе точно 3 удари. Најпрво, во програмата се внесува времето за кое играчот треба да ги нанесе 3-те удари (времето е во секунди). Во табелата се дадени ударите со нивното време на извршување, и колку штета нанесуваат. Ударите се внесуваат преку внесување на бројчето на ударот (првата колона во табелата), притоа се пресметува и се печати на екран после секој удар уште колку време му преостанува на корисникот. Корисникот не смее да преземе удар што ќе го надмине времето. Доколку корисникот внесе удар кој ќе го надмине преостанатото време, тогаш тој удар нанесува штета 0. На крај да се отпечати колку вкупно штета има нанесено корисникот.

Број	Удар	Времетраење	Штета
1	Удар со рака	3 секунди	3
2	Удар со нога	5 секунди	4
3	Магија	10 секунди	8
4	Скок	1 секунда	1
5	Аперкат	7 секунди	6

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

```
int getSekundi(int u){
```

```
    switch (u){
```

```
    case 1: return 3;
```

```
    case 2: return 5;
```

```
    case 3: return 10;
```

```
    case 4: return 1;
```

```
    case 5: return 7;
```

```
    default: return 0;
```

```
    }
```

```
}
```

```
int getShteta(int u){
```

```
    switch (u){
```

```
case 1: return 3;
case 2: return 4;
case 3: return 8;
case 4: return 1;
case 5: return 6;
default: return 0;
}
}

int main(){
int sekundi,shteta;
int u1,u2,u3;
shteta=0;

printf("vnesi vkupen broj na sekundi");
scanf("%d",&sekundi);
printf("vnesi go prvot udar");
scanf("%d",&u1);
sekundi = sekundi - getSekundi(u1);
if(sekundi>=0){
    shteta = shteta + getShteta(u1);
}
printf("vnesi go vtor udar");
scanf("%d",&u2);
sekundi = sekundi - getSekundi(u2);
if(sekundi>=0){
    shteta = shteta + getShteta(u2);
}
printf("vnesi go tret udar");
scanf("%d",&u3);
sekundi = sekundi - getSekundi(u3);
```

```

if(sekundi>=0){
    shteta = shteta + getShteta(u3);
}

printf("vkupno nanesena shteta %d \n",shteta);

return 0;
}

```

2. Да се напише програма во која од тастатура се внесуваат почеток и крај на интервал на цели броеви. Програмата треба да ги испечати сите **силни** броеви од интервалот. Силен број е оној број кој што е еднаков на сумата на факториелите на неговите цифри. На крај да се испечати колку такви броеви постојат.

Пример:

145 = 1! + 4! + 5! - овој број е силен // ќе го испечати
 112 = 1! + 1! + 2! - овој број не е силен // нема да го испечати
 #include <stdio.h>

```

int fel(int broj){
    int rez=1;
    while(broj>0){
        rez = rez * broj;
        broj--;
    }
    return rez;
}

int main(){
    int o1,o2;// int od, do;
    int i,suma,cifra,pom;
    printf("Vnesi go opsegot");
    scanf("%d%d",&o1,&o2);
    for(i=o1;i<=o2;i++){
        suma =0;
        pom = ;
        while(pom>0){
            cifra = pom%10;
            suma = suma + fel(cifra);
            pom/=10;
        }
        if(i==suma){
            printf("brojot %d e silen broj",i);
        }
    }
    return 0;
}

```

}

3. Да се напише рекурзивна функција преку која бројот ќе се трансформира на тој начин што секоја непарна цифра ќе се намали за 1, а парните цифри ќе бидат заменети со нула. Функцијата треба да го врати трансформираниот број.

Да се напише и главна програма која ќе ја тестира работата на функцијата.

Пример:

Повикот f(93854) враќа 82040

#include <stdio.h>

```
int transform(int broj){
    if(broj==0)return 0;
    int cifra = broj%10;
    if(cifra %2 == 0){
        cifra = 0;
    }
    else {
        cifra--;
    }
    printf("brojot e %d, cifrata shto ja dodavam e %d\n",broj,cifra);
    return transform(broj/10)*10+cifra;
}
int main(){
    printf("Brojot e %d\n",transform(93854));
    return 0;
}
```