



**4. (2 boda)**

U prostor označen pravokutnikom upišite sadržaj registra u kojem je, prema standardu IEEE 754 za prikaz brojeva u jednostrukoj preciznosti pohranjen broj  $-8.625$ . **Sadržaj registra napišite u oktalnom obliku.**

**5. (2 boda)**

U prostoru označenom pravokutnikom napišite odsječak programa tako da obavlja potpuno istu funkciju kao prikazani odsječak, pri čemu umjesto petlje s poznatim brojem ponavljanja koristi petlju s ispitivanjem uvjeta na kraju.

```
int suma, i, broj;
scanf("%d", &broj);
suma = 0;
for (i = 10; i < broj; i += 2) {
    suma += i;
}
printf("%d", suma);
```

## Rješenja

```
1.  #include <stdio.h>
    #define DO_KOLIKO 10000

    int main(void) {
        unsigned int testiraniBroj;
        int brojacBitova, posmak;
        for (testiraniBroj = 1; testiraniBroj <= DO_KOLIKO; ++testiraniBroj) {
            brojacBitova = 0;
            posmak = 0;
            while (posmak < 32 && brojacBitova < 5) {
                brojacBitova += testiraniBroj >> posmak & 1;
                ++posmak;
            }
            if (brojacBitova == 4) {
                printf("%4d\n", testiraniBroj);
            }
        }
        return 0;
    }
```

bilo bi ispravno i kada bi bitove izdvajali pomoću `testiraniBroj &` (potencije broja 2)

```
2.  #include <stdio.h>
    #define MAXZNAKOVA 100
    int main(void) {
        char ulaz[MAXZNAKOVA + 1], izlaz[MAXZNAKOVA + 1];
        int indexUlaz = 0, indexIzlaz = 0;
        printf("Upisite niz: ");
        gets(ulaz);
        while (ulaz[indexUlaz] != '\0') {
            if (indexIzlaz % 2 == 0) { /* trebam veliko slovo */
                if (ulaz[indexUlaz] >= 'A' && ulaz[indexUlaz] <= 'Z') {
                    izlaz[indexIzlaz] = ulaz[indexUlaz];
                    ++indexIzlaz;
                }
            } else { /* trebam malo slovo */
                if (ulaz[indexUlaz] >= 'a' && ulaz[indexUlaz] <= 'z') {
                    izlaz[indexIzlaz] = ulaz[indexUlaz];
                    ++indexIzlaz;
                }
            }
            ++indexUlaz;
        }
        izlaz[indexIzlaz] = '\0';
        printf("Izlazni niz: %s\n", izlaz);
        printf("Ulazni niz: %s\n", ulaz);
        return 0;
    }
```

3. `#include <stdio.h>`  
`#define MAXRED 5`  
`#define MAXSTUP 10`  
`int main(void) {`  
 `int i, j, sumaStupcaLijevo, najveci;`  
 `int mat[MAXRED][MAXSTUP];`  
 `printf("Upisite clanove polja:\n");`  
 `for (i = 0; i < MAXRED; ++i)`  
 `for (j = 0; j < MAXSTUP; ++j)`  
 `scanf("%d", &mat[i][j]);`  
  
 `printf("Indeksi stupaca:\n", j);`  
 `for (j = 1; j < MAXSTUP; ++j) {`  
 `/* odredi najveci clan u stupcu j, te sumu lijevog susjednog stupca */`  
 `sumaStupcaLijevo = mat[0][j-1];`  
 `najveci = mat[0][j];`  
 `for (i = 1; i < MAXRED; ++i) {`  
 `sumaStupcaLijevo += mat[i][j-1];`  
 `if (mat[i][j] > najveci) {`  
 `najveci = mat[i][j];`  
 `}`  
 `}`  
 `if (najveci == sumaStupcaLijevo) {`  
 `printf("%d\n", j);`  
 `}`  
 `}`  
 `return 0;`  
`}`

4. 3 0 1 0 2 4 0 0 0 0 0

5. `int suma, i, broj;`  
`scanf("%d", &broj);`  
`suma = 0;`  
`i = 10;`  
`do {`  
 `if (i < broj) {`  
 `suma += i;`  
 `i += 2;`  
 `}`  
`} while (i < broj);`  
`printf("%d", suma);`

**ili**

`int suma, i, broj;`  
`scanf("%d", &broj);`  
`suma = 0;`  
`i = 10;`  
`if (i < broj) {`  
 `do {`  
 `suma += i;`  
 `i += 2;`  
 `} while (i < broj);`  
`}`  
`printf("%d", suma);`