

ZŁOŻONOŚĆ OBLICZENIOWA

ZADANIA

Z1. Oszacuj złożoność czasową najgorszego przypadku następującego algorytmu sortowania:

```
#define MaxSize 100
#define NONE -1
#include <stdio.h>
int S[MaxSize], Last=-1;
int IndexMax(void) {
    int i, Max;
    Max= 0;
    for(i=1; i<=Last; i++)
        if(S[Max] < S[i]) Max= i;
    return Max;
}
int main(void) {
    int i, Max, Indx;
    for(; scanf("%d", &S[++Last]) != EOF;)
        ;
    for(i=1; i <= Last; i++) {
        Max= S[IndexMax()];
        S[IndexMax]= NONE;
        printf("%d, ", Max);
    }
    return 0;
}
```

Z2. Jaką złożoność ma rekurencyjna wersja algorytmu Fibonacciego?

```
int f (int n) {
    if (n <= 1)
        return 1;
    else
        return f(n-1)+f(n-2);
}
```

Z3. Oszacuj złożoność następującej funkcji:

```
#define MaxN 20
int f (int n) {
    int v[MaxN], i;
    v[0]= v[1]= 1;
    for (i=2; i<=n; i++)
        v[i]=v[i-1]+v[i-2];
    return v[n];
}
```

Z4. Korzystając z notacji O oszacuj liczbę operacji mnożenia następującej wersji schematu Hornera:

```
int p(int x, int n, int a[ ]){
    if (n == 0) return a[0];
    else return p(x, n-1, a)*x + a[n];
}
```