

## Zadanie E: Układy równań

Do dokładnego rozwiązywania układów równań liniowych na ogół stosuje się metodę Gaussa. Z uwagi na niedokładność reprezentacji i obliczeń, wyniki uzyskiwane z użyciem jej podstawowej wersji mogą nie być dokładne. Dlatego rozważa się wiele jej modyfikacji związanych wyborem elementów głównych, a także stosuje się iteracyjne poprawianie wyniku.

Twoim zadaniem jest napisanie w C++ metody rozwiązywania układów równań liniowych postaci:

$$Ax = b,$$

gdzie  $A \in \mathbb{R}^{n \times n}$ ,  $x \in \mathbb{R}^n$ ,  $b \in \mathbb{R}^n$  oraz  $n \leq N = 2000$ .

Omawiana metoda o sygnaturze:

```
void solveEquations(  
    double A[N][N],  
    int n,  
    double b[N],  
    double eps,  
    double x[N]  
);
```

jest połączeniem metody Gaussa ze skalowanym wyborem elementów głównych oraz poprawianiem iteracyjnym. Skale powinny być wyznaczane na początku algorytmu. Funkcja powinna uzupełnić wektor  $x$  rozwiązaniem układu równań z dokładnością  $eps$  (liczonej według normy maksimum), to znaczy:

$$\max\{|r_i(x)| : i = 1, \dots, n\} < eps,$$

gdzie  $r(x) = Ax - b = (r_i(x))_{i=1, \dots, n}$  jest wektorem rezydualnym. Aby zapewnić odpowiednią dokładność obliczeń wartość wektora rezydualnego  $r(x)$  powinna być obliczana w podwójnej precyzji. Zakładamy, że podany układ ma dokładnie jedno rozwiązanie.

Na BaCę należy przesłać plik source.cpp zawierający funkcję solveEquations i ewentualne funkcje pomocnicze ale nie zawierający funkcji main. Stałą  $N = 2000$  typu int zdefiniuj w pliku source.cpp.

Macierze dla przykładowych testów znajdują się w dodatkowych plikach: test1.in oraz test2.in. Pierwsza linia to wymiar macierzy, w kolejnych znajdują się elementy macierzy  $A$ , następnie wektora  $b$ , a w ostatniej dokładność jakiej oczekujemy podając rozwiązanie. Nadesłany kod będzie testowany wykorzystując następującą metodę main:

```
#include <iostream>  
const int N = 2000;  
#include "source.cpp"  
  
using namespace std;  
  
int main(){  
    double (*a)[N] = new double [N][N];  
    double b[N];  
    double x[N];  
    int n;  
    double eps;
```

```

cin >> n;
for(int i = 0; i < n; ++i){
    for(int j = 0; j < n; ++j){
        cin >> a[i][j];
    }
}
for(int i = 0; i < n; ++i){
    cin >> b[i];
}
cin >> eps;

solveEquations(a, n, b, eps, x);

delete [] a;

return 0;
}

```