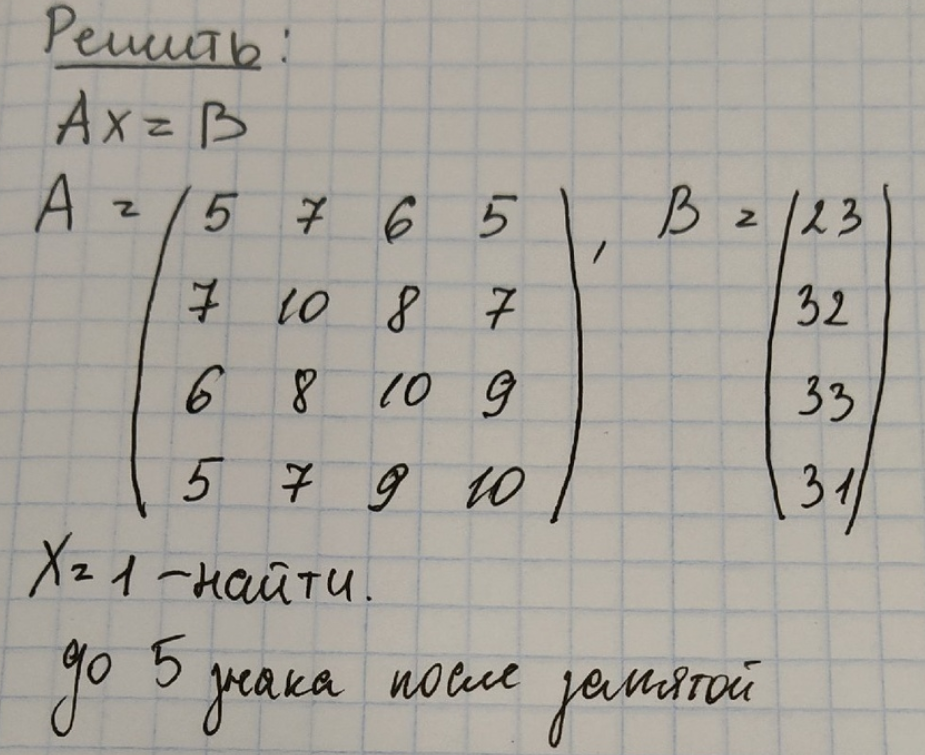
**Лабораторная работа №3. Метод Гаусса (выбор по столбцу)**

1. Код лабораторной работы написан на языке: С;

Компилятор: онлайн среда Repl.it (clang version 7.0.0)

1. Постановка задачи: Составить программу, которая реализует метод Гаусса для решение систем линейных уравнений.



1. Код: <https://repl.it/@sonyadk/LabaVibor2>

| /\*Поиск максимального элемента в столбце\*/  #include <stdlib.h>  #include <stdio.h>  #include <math.h>  const int n = 4;  float a[n][n+1] = {5, 7, 6, 5, 23, 7, 10, 8, 7, 32, 6, 8, 10, 9, 33, 5, 7, 9, 10, 31};  float x[n];  int ind, count[n];  float m, sum, at;  //функция нахождения индекса максимального элемента столбца  float max(int sizeI){  float y = a[sizeI][sizeI];  int index = sizeI;  for (int i = sizeI; i<n; i++){  if (fabs(y) < fabs(a[i][sizeI])){  y = a[i][sizeI]; //запоминаем наибольший элемент  index = i; //запоминаем индекс  }  }  return index;  }  int main(void){  for (int i=0; i<n;i++){  count[i]=i;  }  for (int i = 0; i < n; i++){  //ищем индекс строки максимального элемента столбца  ind = max(i); //запоминаем индекс строки максимального элемента  //меняем местами строки  for (int k=0; k<=n; k++){  m = a[ind][k];  a[ind][k]=a[i][k];  a[i][k]=m;  }  m = count[i];  count[i]=ind;  count[ind]=m;  //этап прямого хода методом оптимальных исключений (матрицу под ведущей строкой преобразуем)  for (int k = i+1; k<=n; k++){  at = a[k][i]/a[i][i];  for(int c = i; c<=n+1; c++ ){  a[k][c]=a[k][c]-at\*a[i][c];  }  }  }  //находим x как и в методе отпимальных исключений  for (int i = n-1; i>=0; i--){  x[i] = 0;  sum = 0;  for (int j = i+1; j<n; j++){  sum += a[i][j]\*x[j];  }  x[i] = (a[i][n]-sum)/a[i][i];  }  //выводим преобразованную матрицу и значения x  for (int i = 0; i<n; i++){  printf("\n");  for (int j = 0; j<=n; j++){  printf("%7.2f ",a[i][j]);  }  printf("| x%d = %5.5f", i+1, x[count[i]]);  }  return 0;  } |
| --- |

1. Результат:  
   