Бахтибекова Салима Шерозовна

3ПКС-320

Экзаменационный билет №2

Цель: выполнить консольный проект на C# с применением кода на assembler.

Листинг программы, выполненной на C#:

|  |
| --- |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using System.IO;  namespace Examen  {  class sotrudnik //класс сотрудник  {  public string subname; //характеристика "фамилия"  public List<int> zarpl = new List<int>(); //характеристика "коллекция зарплат по месяцам"  public sotrudnik() { }  public sotrudnik(string sn) { subname = sn; } //способ создания класса вида "название\_класса(фамилия)"  public sotrudnik(string sn, List<int> m) { } //способ создания класса вида "название\_класса(фамилия, зарплаты)"  public void infa() //метод для вывода на экран информации о зарплатах  {  Console.Write("Фамилия сотрудника: " + subname + "; Зарплата: ");  for (int i = 0; i < zarpl.Count; i++)  {  if (i == zarpl.Count - 1) { Console.Write(zarpl[i] + "."); }  else { Console.Write(zarpl[i] + ", "); }  }  Console.WriteLine();  }  }  class Program  {  static void Main(string[] args)  {  Random rnd = new Random();  int a = rnd.Next(1, 11); //генерация количества зарплат  int b = rnd.Next(40000, 60001); //генерация зарплаты  string tofile = "";  sotrudnik[] sotrudniks = new sotrudnik[] { new sotrudnik("Костиков"), new sotrudnik("Снежная"), new sotrudnik("Дилеева") }; //коллекция с 3 объектами класса  for (int i = 0; i < sotrudniks.Length; i++)  {  for (int j = 0; j < a; j++)  {  sotrudniks[i].zarpl.Add(b);  b = rnd.Next(40000, 60001);//генерация зарплат  }  a = rnd.Next(1, 11);//генерация количества зарплат  }  for (int i = 0; i < sotrudniks.Length; i++)  {  sotrudniks[i].infa();  }  //пример кода, выполненного в ассемблере, для вывода на консоль в программе c#  double[] summa = new double[3]{ 0,0,0};  double[] sr\_zp = new double[3];  int sch = 0;  for (int i = 0; i < sotrudniks.Length; i++)  {  for (int j = 0; j < sotrudniks[i].zarpl.Count; j++)  {  summa[i] = summa[i] + sotrudniks[i].zarpl[j];  sch = sch+1;  }  sr\_zp[i] = summa[i] / sch;  sch = 0;  }  Math.Round(sr\_zp[0], 2);  for (int i = 0; i < sotrudniks.Length; i++)  {  Console.WriteLine("Сумма зарплат: " + summa[i] + ", средняя зарплата: " + Math.Round(sr\_zp[i], 2));  }  //заполнение переменной типа string информацией, которую необходимо вывести в файл.txt  for (int i = 0; i < sotrudniks.Length; i++)  {  if (sotrudniks[i].zarpl.Count >= 6)  {  tofile = tofile + sotrudniks[i].subname + ", зарплата за 6 месяцев: " + sotrudniks[i].zarpl[0] + " " + sotrudniks[i].zarpl[1] + " " + sotrudniks[i].zarpl[2] + " " + sotrudniks[i].zarpl[3] + " " + sotrudniks[i].zarpl[4] + " " + sotrudniks[i].zarpl[5] + ", средняя зарплата " + Math.Round(sr\_zp[i], 2) + "\n";  }  else {  tofile = tofile + sotrudniks[i].subname + ", зарплата за 6 месяцев: ";  for (int j = 0; j < sotrudniks[i].zarpl.Count; j++)  {  tofile = tofile + sotrudniks[i].zarpl[j] + " ";  }  tofile = tofile + ", средняя зарплата " + Math.Round(sr\_zp[i], 2) + "\n";  }  }  string path = @"C:\Users\tinag\OneDrive\Рабочий стол\дз\экзамен.txt";//указание пути к файлу, в который идёт вывод информации  File.WriteAllText(path, tofile);//запись информации в файл  string name = File.ReadAllText(path);//чтение информации с файла  Console.WriteLine(name);//вывод в консоль информации из файла  Console.ReadKey(true);  }  }  } |

Листинг программы, выполненной отдельной задачей во flat assembler:

|  |
| --- |
| format PE console  include 'win32a.inc'  start:  ;; макрос для вывода на экран зарплат  macro prntn op1,op2  {  local metka  mov esi,op1  mov ecx,op2  mov eax,0  metka:  mov eax,[esi]  push ecx  cinvoke printf,fmat,eax  pop ecx  add esi,4  loop metka  }  cinvoke printf, text2  ;;массив зарплат, вводимый в теле программы  lea ebx,[A]  mov [ebx], dword 48874  mov [ebx+4],dword 58527  mov [ebx+8],dword 49138  mov [ebx+12],dword 52062  mov [ebx+16],dword 57151  mov [ebx+20],dword 59026  mov [ebx+24],dword 46333  mov [ebx+28],dword 43649  mov [ebx+32],dword 46633  ;;вывод на экран зарплат костикова  prntn A,[N]  ;;вызов процедуры для нахождения средней зарплаты костикова  lea eax,[A]  push eax  mov eax,[N]  push eax  call average  mov ebx,eax  cinvoke printf, text3,eax  cinvoke printf, text  cinvoke printf, text5  ;; массив зарплат, вводимый в теле программы  lea ebx,[B]  mov [ebx], dword 49031  mov [ebx+4],dword 46370  mov [ebx+8],dword 52841  mov [ebx+12],dword 43236  mov [ebx+16],dword 57473  mov [ebx+20],dword 46342  mov [ebx+24],dword 56241  mov [ebx+28],dword 58512  mov [ebx+32],dword 51779  mov [ebx+36],dword 43422  ;; ;;вывод на экран зарплат снежной  prntn B,[NN]  ;; вызов процедуры для нахождения средней зарплаты снежной  lea eax,[B]  push eax  mov eax,[NN]  push eax  call average  mov ebx,eax  cinvoke printf, text3,eax  cinvoke printf, text  cinvoke printf, text6  ;; массив зарплат, вводимый в теле программы  lea ebx,[C]  mov [ebx], dword 48968  mov [ebx+4],dword 50559  mov [ebx+8],dword 50900  mov [ebx+12],dword 48150  mov [ebx+16],dword 45444  mov [ebx+20],dword 46944  ;; ;;вывод на экран зарплат дилеевой  prntn C,[NNN]  ;; вызов процедуры для нахождения средней зарплаты дилеевой  lea eax,[C]  push eax  mov eax,[NNN]  push eax  call average  mov ebx,eax  cinvoke printf, text3,eax  invoke sleep,1000000  invoke exit,0  ;;процедура для 1)расчёта сумм зарплата и 2)расчёта средней зарплаты  proc average  push ebp  mov ebp, esp  push ebx  push ecx  mov ecx, [ebp+8]  mov ebx, [ebp+12]  xor eax,eax  xor edi,edi  jcxz m4  cycln:  cmp [ebx],esi  jg m5  inc edi  add eax,[ebx]  m5:  add ebx,4  loop cycln  cdq  idiv edi  m4:  pop ecx  pop ebx  pop ebp  ret 8  endp  ;;выделение места для массивов зарплат  A dd 9 dup(?)  N dd 9  B dd 10 dup(?)  NN dd 10  C dd 6 dup(?)  NNN dd 6  text db 10, 13, '', 10, 13, 0  text2 db 10,13,'Kostikov ',0  text5 db 10,13,'Snejnaya ',0  text6 db 10,13,'Dileeva ',0  text3 db 10,13,'Average : %d ',0  text4 db 10,13,'kachestvo znani: %d ',0  fmat db '%d',9,0  spp db '%d',0  ; import data in the same section  data import  library msvcrt,'MSVCRT.DLL',\  kernel32,'KERNEL32.DLL'  import kernel32,\  sleep,'Sleep'  import msvcrt,\  printf,'printf',\  scanf,'scanf',\  exit,'exit',\  setlocale,'setlocale',\  system,'system'  end data |

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1 Результат программы, выполненной на C#

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2 Результат вывода в файл фамилий сотрудников, зарплат за 6 месяцев и средних зарплат

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3 Результат задачи, выполненной во flat assembler

Вывод: Мне удалось выполнить консольный проект на C# с применением кода на assembler.