**Вариант 15**

**Задача:** По результатам обследования винодельческих заводов выбрать предприятия с объемом выпуска товарной продукции, лежащем в заданном интервале (задается граничными значениями ). Для каждого выбранного завода определить производительность труда и фондоотдачу. Количество предприятий не более 20. Результаты распечатать в виде таблицы с шапкой:

Предприятия с объемом выпуска товарной продукции

в интервале от .... до .... (млн. руб.)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| №  п/п | Район | Предприятие | Товарная продукция  (млн. руб.) | Стоимость основных производственных фондов  (млн. руб.) | Численность работающих | Производительность труда  (руб.) | Фондоотдача  (руб. на руб. затрат) |
| ? | 9 сим | 16 сим. | цц.ццц | цц.ццц | цццц | ? | ? |
| Итого (среднее) | | | | | | ? | ? |

Предприятие (наименование):

- с наибольшей фондоотдачей  
- с наибольшей производительностью труда

**Указания:** Заполнить исходную таблицу и отсортировать ее по возрастанию производительности труда. После этого в цикле, пока не откажется пользователь

* запрашивать верхнюю и нижнюю границы выпуска товарной продукции (млн.руб.),
* копировать из исходной в рабочую таблицу строки с товарной продукцией (млн.руб.), принадлежащей
* заданному интервалу,
* выявлять предприятие с наибольшей фондоотдачей и с наибольшей производительностью труда,
* выдавать сведения об объеме выпуска товарной продукции.

**Математическая постановка задачи:**

Дано:

tab(i)=<tab(i,1), tab(i, 2), tab(i,3), tab(i,4), tab(i,5), tab(i,6), tab(i,7), tab(i,8)>, где

tab(i,1) – номер

tab(i,2) – район

tab(i,3) – предприятие

tab(i,4) – товарная продукция

tab(i,5) – стоимость основных производственных фондов

tab(i,6) – численность работающих

tab(i,7) – производительность труда

tab(i,8) – фондоотдача

tab(i,9) – объем выпуска товарной продукции

Найти:

tab(i, 7) = tab(i,9) / tab(i,6)

tab(i, 8) = tab(i,9) / tab(i,5)

tab(i, 9) = tab(i,4) + tab(i,5)

itogo1 =

itogo2 =   
  
Контрольные примеры:  
  
№1

Вводные данные:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Район | Предприятие | Товарная продукция  (млн. руб.) | Стоимость основных производственных фондов  (млн. руб.) | Численность работающих |
| Северный | Эко продукты | 45.749 | 15.194 | 4128 |
| Западный | ООО ГМО | 23.234 | 20.234 | 5000 |
| Восточный | ТоварНО | 89.456 | 30.864 | 356 |
| Южный | ВкусХЭЛ | 52.578 | 32.374 | 2500 |

Изначальный результат программы:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| №  п/п | Район | Предприятие | Товарная продукция  (млн. руб.) | Стоимость основных производственных фондов  (млн. руб.) | Численность работающих | Производительность труда  (руб.) | Фондоотдача  (руб. на руб. затрат) |
| 1 | Северный | Эко продукты | 45.749 | 15.194 | 4128 | 0.0147 | 4.0109 |
| 2 | Западный | ООО ГМО | 23.234 | 20.234 | 5000 | 0.0086 | 2.1482 |
| 3 | Восточный | ТоварНО | 89.456 | 30.864 | 356 | 0.3379 | 3.8983 |
| 4 | Южный | ВкусХЭЛ | 52.578 | 32.374 | 2500 | 0.0339 | 2.6240 |

После сортировки:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| №  п/п | Район | Предприятие | Товарная продукция  (млн. руб.) | Стоимость основных производственных фондов  (млн. руб.) | Численность работающих | Производительность труда  (руб.) | Фондоотдача  (руб. на руб. затрат) |
| 3 | Восточный | ТоварНО | 89.456 | 30.864 | 356 | 0.3379 | 3.8983 |
| 4 | Южный | ВкусХЭЛ | 52.578 | 32.374 | 2500 | 0.0339 | 2.6240 |
| 1 | Северный | Эко продукты | 45.749 | 15.194 | 4128 | 0.0147 | 4.0109 |
| 2 | Западный | ООО ГМО | 23.234 | 20.234 | 5000 | 0.0086 | 2.1482 |

Итог:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| №  п/п | Район | Предприятие | Товарная продукция  (млн. руб.) | Стоимость основных производственных фондов  (млн. руб.) | Численность работающих | Производительность труда  (руб.) | Фондоотдача  (руб. на руб. затрат) |
| 3 | Восточный | ТоварНО | 89.456 | 30.864 | 356 | 0.3379 | 3.8983 |
| 4 | Южный | ВкусХЭЛ | 52.578 | 32.374 | 2500 | 0.0339 | 2.6240 |
| 1 | Северный | Эко продукты | 45.749 | 15.194 | 4128 | 0.0147 | 4.0109 |
| 2 | Западный | ООО ГМО | 23.234 | 20.234 | 5000 | 0.0086 | 2.1482 |
| Итого (среднее) | | | | | | 0.0988 | 3.1704 |

№2

Вводные данные:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Район | Предприятие | Товарная продукция  (млн. руб.) | Стоимость основных производственных фондов  (млн. руб.) | Численность работающих |
| С | Эко продукты | 45.749 | 15.194 | 4128 |
| З | ООО ГМО | 23.234 | 20.234 | 5000 |
| В | ТоварНО | 89.456 | 30.864 | 356 |
| Ю | ВкусХЭЛ | 52.578 | 32.374 | 2500 |
| СВ | Паёк | 11.294 | 5.946 | 76 |
| СЗ | Печенье | 96.214 | 79.746 | 3157 |
| ЮВ | Моржэ | 33.738 | 15.277 | 1443 |
| ЮЗ | ОРН | 19.025 | 18.575 | 500 |

Изначальный результат программы:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| №  п/п | Район | Предприятие | Товарная продукция  (млн. руб.) | Стоимость основных производственных фондов  (млн. руб.) | Численность работающих | Производительность труда  (руб.) | Фондоотдача  (руб. на руб. затрат) |
| 1 | С | Эко продукты | 45.749 | 15.194 | 4128 | 0.0147 | 4.0109 |
| 2 | З | ООО ГМО | 23.234 | 20.234 | 5000 | 0.0086 | 2.1482 |
| 3 | В | ТоварНО | 89.456 | 30.864 | 356 | 0.3379 | 3.8983 |
| 4 | Ю | ВкусХЭЛ | 52.578 | 32.374 | 2500 | 0.0339 | 2.6240 |
| 5 | СВ | Паёк | 11.294 | 5.946 | 76 | 0.2268 | 2.8994 |
| 6 | СЗ | Печенье | 96.214 | 79.746 | 3157 | 0.0557 | 2.2065 |
| 7 | ЮВ | Моржэ | 33.738 | 15.277 | 1443 | 0.0339 | 3.2084 |
| 8 | ЮЗ | ОРН | 19.025 | 18.575 | 500 | 0.0752 | 2.0242 |

После сортировки:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| №  п/п | Район | Предприятие | Товарная продукция  (млн. руб.) | Стоимость основных производственных фондов  (млн. руб.) | Численность работающих | Производительность труда  (руб.) | Фондоотдача  (руб. на руб. затрат) |
| 3 | В | ТоварНО | 89.456 | 30.864 | 356 | 0.3379 | 3.8983 |
| 5 | СВ | Паёк | 11.294 | 5.946 | 76 | 0.2268 | 2.8994 |
| 8 | ЮЗ | ОРН | 19.025 | 18.575 | 500 | 0.0752 | 2.0242 |
| 6 | СЗ | Печенье | 96.214 | 79.746 | 3157 | 0.0557 | 2.2065 |
| 4 | Ю | ВкусХЭЛ | 52.578 | 32.374 | 2500 | 0.0339 | 2.6240 |
| 7 | ЮВ | Моржэ | 33.738 | 15.277 | 1443 | 0.0339 | 3.2084 |
| 1 | С | Эко продукты | 45.749 | 15.194 | 4128 | 0.0147 | 4.0109 |
| 2 | З | ООО ГМО | 23.234 | 20.234 | 5000 | 0.0086 | 2.1482 |

Итого:

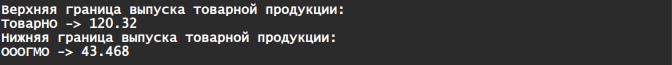
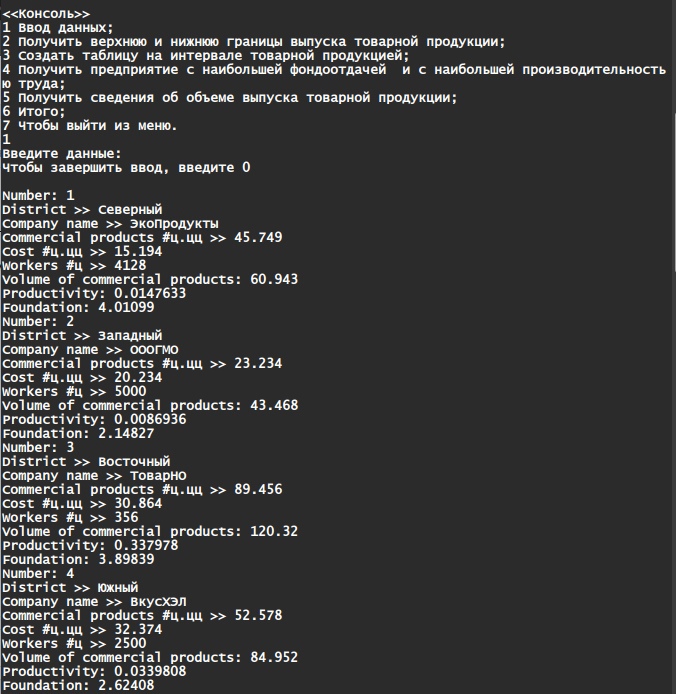
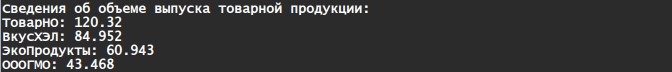
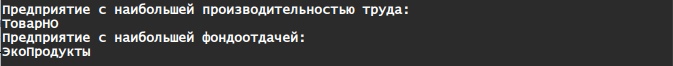
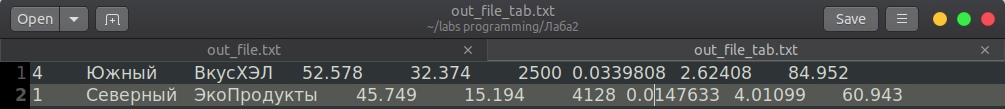
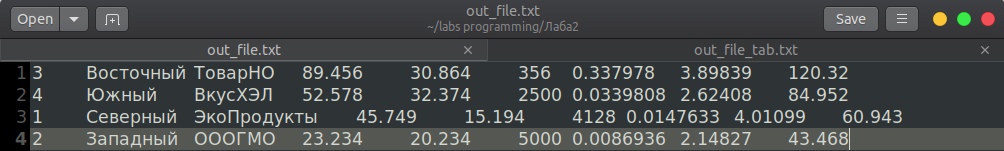
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| №  п/п | Район | Предприятие | Товарная продукция  (млн. руб.) | Стоимость основных производственных фондов  (млн. руб.) | Численность работающих | Производительность труда  (руб.) | Фондоотдача  (руб. на руб. затрат) |
| 3 | В | ТоварНО | 89.456 | 30.864 | 356 | 0.3379 | 3.8983 |
| 5 | СВ | Паёк | 11.294 | 5.946 | 76 | 0.2268 | 2.8994 |
| 8 | ЮЗ | ОРН | 19.025 | 18.575 | 500 | 0.0752 | 2.0242 |
| 6 | СЗ | Печенье | 96.214 | 79.746 | 3157 | 0.0557 | 2.2065 |
| 4 | Ю | ВкусХЭЛ | 52.578 | 32.374 | 2500 | 0.0339 | 2.6240 |
| 7 | ЮВ | Моржэ | 33.738 | 15.277 | 1443 | 0.0339 | 3.2084 |
| 1 | С | Эко продукты | 45.749 | 15.194 | 4128 | 0.0147 | 4.0109 |
| 2 | З | ООО ГМО | 23.234 | 20.234 | 5000 | 0.0086 | 2.1482 |
| Итого (среднее) | | | | | | 0.0983 | 2.8775 |

Код программы:

|  |
| --- |
| #include <cstdlib> //заголовок для стандартной библиотеки С  #include <iostream> //заголовок для поддержки системы ввода-вывода в С++  #include <iomanip>  #include <fstream>  using namespace std;  ofstream f;  ofstream c;  struct DATA\_FACTORY {  int number;  string district;  string company\_name;  float commercial\_products;  float cost;  int workers;  float productivity;  float foundation;  float volume\_of\_commercial\_products;  };  const int n = 20; //Array size  DATA\_FACTORY data\_array[n];  void data\_input();  void get\_up\_and\_down();  void create\_tab();  void max\_company();  void show\_volume\_of\_commercial\_products();  void sorting();  void itog();  void (\*menu[7])() = { data\_input, get\_up\_and\_down, create\_tab, max\_company, show\_volume\_of\_commercial\_products, itog, sorting };  int main(){  cout << "\n<<Консоль>>";  cout << "\n1 Ввод данных;";  cout << "\n2 Получить верхнюю и нижнюю границы выпуска товарной продукции;";  cout << "\n3 Создать таблицу на интервале товарной продукцией;";  cout << "\n4 Пффолучить предприятие с наибольшей фондоотдачей и с наибольшей производительностью труда;";  cout << "\n5 Получить сведения об объеме выпуска товарной продукции;";  cout << "\n6 Итого;";  cout << "\n7 Чтобы выйти из меню.\n";  while(1){  cout << "\n>> ";  int m;  cin >> m;  cout << endl;  if (m == 7) break;  menu[m-1]();  }  return 0;  }  void data\_input(){  cout << "Введите данные:\n";  cout << "Чтобы завершить ввод, введите 0\n\n";    for (int i = 0; i < n; i++){  if (data\_array[i].number == NULL){  if (data\_array[i].number == i+1) break;  cout << "Number: " << i+1 << endl;  cout << "District >> ";  cin >> data\_array[i].district;  if (data\_array[i].district == "0") break;  data\_array[i].number = i+1;  cout << "Company name >> ";  cin >> data\_array[i].company\_name;  if (data\_array[i].company\_name == "0") break;  cout << "Commercial products #ц.цц >> ";  cin >> data\_array[i].commercial\_products;  if (data\_array[i].commercial\_products == 0) break;  cout << "Cost #ц.цц >> ";  cin >> data\_array[i].cost;  if (data\_array[i].cost == 0) break;  cout << "Workers #ц >> ";  cin >> data\_array[i].workers;  if (data\_array[i].workers == 0) break;  data\_array[i].volume\_of\_commercial\_products = data\_array[i].commercial\_products + data\_array[i].cost; //Объём выпуска товарной продукции  cout << "Volume of commercial products: " << data\_array[i].volume\_of\_commercial\_products << "\n";  data\_array[i].productivity = data\_array[i].volume\_of\_commercial\_products / data\_array[i].workers;  cout << "Productivity: " << data\_array[i].productivity << "\n";  data\_array[i].foundation = data\_array[i].volume\_of\_commercial\_products / data\_array[i].cost;  cout << "Foundation: " << data\_array[i].foundation << "\n";  f << data\_array[i].number << "\t" << data\_array[i].district << "\t" << data\_array[i].company\_name << "\t" << data\_array[i].commercial\_products << "\t" << data\_array[i].cost << "\t" << data\_array[i].workers << "\t" << data\_array[i].productivity << "\t" << data\_array[i].foundation << "\t" << data\_array[i].volume\_of\_commercial\_products << "\n";  }  }  menu[6]();  }  void sorting(){  f.open("out\_file.txt", ios::out );  for (int i = 0; i < n; i++){  for (int j = i+1; j < n-1; j++){  if (data\_array[i].number == NULL or data\_array[j].number == NULL) break;  if (data\_array[i].productivity < data\_array[j].productivity){  swap(data\_array[i].number, data\_array[j].number);  swap(data\_array[i].district, data\_array[j].district);  swap(data\_array[i].company\_name, data\_array[j].company\_name);  swap(data\_array[i].commercial\_products, data\_array[j].commercial\_products);  swap(data\_array[i].cost, data\_array[j].cost);  swap(data\_array[i].workers, data\_array[j].workers);  swap(data\_array[i].productivity, data\_array[j].productivity);  swap(data\_array[i].foundation, data\_array[j].foundation);  swap(data\_array[i].volume\_of\_commercial\_products, data\_array[j].volume\_of\_commercial\_products);  }  }  }  for (int i = 0; i < n; i++){  if (data\_array[i].number == NULL) break;  cout << data\_array[i].number << endl;  cout << data\_array[i].district << endl;  cout << data\_array[i].company\_name << endl;  cout << data\_array[i].commercial\_products << endl;  cout << data\_array[i].cost << endl;  cout << data\_array[i].workers << endl;  cout << data\_array[i].productivity << endl;  cout << data\_array[i].foundation << endl;  cout << data\_array[i].volume\_of\_commercial\_products << endl;  f << data\_array[i].number << "\t" << data\_array[i].district << "\t" << data\_array[i].company\_name << "\t" << data\_array[i].commercial\_products << "\t" << data\_array[i].cost << "\t" << data\_array[i].workers << "\t" << data\_array[i].productivity << "\t" << data\_array[i].foundation << "\t" << data\_array[i].volume\_of\_commercial\_products << "\n";  }  f.close();  }  void get\_up\_and\_down(){  float max\_volume = 0.0;  int max\_i;  int min\_i;  float min\_volume = 100000000000.0;  for (int i = 0; i < n; i++){  if (data\_array[i].number == NULL) break;  if (data\_array[i].volume\_of\_commercial\_products < min\_volume){  min\_volume = data\_array[i].volume\_of\_commercial\_products;  min\_i = i;  }  if (data\_array[i].volume\_of\_commercial\_products > max\_volume){  max\_volume = data\_array[i].volume\_of\_commercial\_products;  max\_i = i;  }  }  cout << "\nВерхняя граница выпуска товарной продукции:\n" << data\_array[max\_i].company\_name << " -> " << data\_array[max\_i].volume\_of\_commercial\_products;  cout << "\nНижняя граница выпуска товарной продукции:\n" << data\_array[min\_i].company\_name << " -> " << data\_array[min\_i].volume\_of\_commercial\_products << endl;  }  void create\_tab(){  c.open("out\_file\_tab.txt", ios::out );  float interval\_ot, interval\_do;  cout << "Введите интервал:\n";  cout << "От >> "; cin >> interval\_ot;  cout << "До >> "; cin >> interval\_do;  cout << endl;  for (int i = 0; i < n; i++){  if (data\_array[i].number == NULL) break;  if ((data\_array[i].commercial\_products < interval\_do) and (data\_array[i].commercial\_products > interval\_ot) ){  c << data\_array[i].number << "\t" << data\_array[i].district << "\t" << data\_array[i].company\_name << "\t" << data\_array[i].commercial\_products << "\t" << data\_array[i].cost << "\t" << data\_array[i].workers << "\t" << data\_array[i].productivity << "\t" << data\_array[i].foundation << "\t" << data\_array[i].volume\_of\_commercial\_products << "\n";  }    }  c.close();  }  void max\_company(){  int i\_p;  int i\_f;  float max\_productivity = 0.0;  float max\_foundation = 0.0;  for (int i = 0; i < n; i++){  if (data\_array[i].number == NULL) break;  if (data\_array[i].productivity > max\_productivity){  max\_productivity = data\_array[i].productivity;  i\_p = i;  }  if(data\_array[i].foundation > max\_foundation){  max\_foundation = data\_array[i].foundation;  i\_f = i;  }    }  cout << "Предприятие c наибольшей производительностью труда:\n" << data\_array[i\_p].company\_name << endl;  cout << "Предприятие c наибольшей фондоотдачей:\n" << data\_array[i\_f].company\_name << endl;  }  void show\_volume\_of\_commercial\_products(){  cout << "Сведения об объеме выпуска товарной продукции:\n";  for (int i = 0; i < n; i++){  if (data\_array[i].number == NULL) break;  cout << data\_array[i].company\_name << ": " << data\_array[i].volume\_of\_commercial\_products << endl;  }  }  void itog(){  float volume\_productivity = 0.0;  float volume\_foundation = 0.0;  cout << "Итого:\n";  int j = 0;  for (int i = 0; i < n; i++){  if (data\_array[i].number == NULL) break;  volume\_productivity = volume\_productivity + data\_array[i].productivity;  volume\_foundation = volume\_foundation + data\_array[i].foundation;  j++;  }  cout << "Productivity: " << volume\_productivity / j << " Foundation: " << volume\_foundation / j << endl;  } |

Результат работы программы:

№1

https://sun9-42.userapi.com/impg/Cvc8Fvl6o-bOD8sddkmKimWMlwSmnW7NoDetcg/VVptNBeNPgg.jpg?size=672x46&quality=96&sign=8bd5d734a9d200ca6217c13de34bceec&type=albumhttps://sun9-31.userapi.com/impg/LbUjCTqhRt_XNyvCjkLYef80dqGbHVBHGzfu4g/WVVfP2_gofA.jpg?size=668x60&quality=96&sign=3e9d0d8971fa374156668255e2d1103e&type=album

№2

