# Линейный алгоритм

## Задание 1

/\*На двух кустах сидело 16 птиц

со второго куста улетело две птицы,

а затем с первого куста на второй перелетели 5 птиц

после этого на каждом кусте оказалось одно и тоже число птиц.

Вопрос, сколько птиц было в начале на каждом кусте?\*/

**void** **task1**(){

**int** x = (**16** + **8**)/**2**;

**int** y = **16** - x;

cout << x << " " << y << endl;

}

## Задание 2

/\* Велосипедист (В) должен попасть в место назначения к определённому сроку.

Известно, что если он поедет со скоростью 15 км в час, то приедет на час раньше,

а если скорость будет 10 км в час, то опоздает на 1 час.

С какой скоростью должен ехать велосипедист, чтобы приехать вовремя?\*/

/\*

Дано:

v1 = 10 км/ч, на 1 час позже

v2 = 15 км/ч, на 1 час раньше

Найти:

v3 - оптимальная скорость

Рассуждение:

1. Движение со скоростью v1 вызовет опоздание на 1 час,

Соответственно, в таком случае В будет находится

в 1 часе езды до места, или же в 10 км

2. При движении со скоростью v2 приведёт к опережению событий,

и если бы В продолжил путь, он бы к сроку преодолел лишние 15 км

3. Предположим, что В обдумывая данную поездку, сперва мысленно проехал со скоростью v1. Обнаружил что ему недостаточно времени, и мысленно повторил маршрут со скоростью v2.

4. В таком случае, v1 и v2 связаны посредством умозаключений В.

И, предположим, прослеживается в том, что В после первой неудачной попытки, намеренно или случайно повысил v1 на 50%

5. В следствии чего он получил, повышение скорости на 50% повысило пройденную дистанцию на 25 км, т.е каждые 2% прибавки скорости, дают ему 1 км пути

6. Вспомним, что для того что б прибыть в место назначения к необходимому времени он должен повысить свою скорость так, чтобы преодолеть за то же время на 10 км больше.

7. И теперь вычислим требуемую скорость: Для преодоления недостающих 10 км, В должен повысить свою скорость на 20% (10\*2%), что равно: 2 км/ч (10\*0,2).

8. Значит, оптимальной скоростью является скорость в 12 км/ч (10 + 2 км/ч)

\*/

**void** **task2**(){

**int** v1 = **10**, v2 = **10**;

**int** s1 = **1**, s2 = **1**, so;

**double** p, km, v, v3;

**float** pv1;

s1 = s1 \* v1; s2 = s2 \* v2; // растояния которого нам не хватило/которое мы переехали

pv1 = v1/**100.0**; // чему равен 1 процент от v1

so = s1 + s2; // дистанция которую мы проехали повысив скорость

p = (v2 - v1) / pv1; // на сколько % повысили скорость

km = p / so; // сколько % рывны 1 км

v = s1 \* km / **100.0**;

v3 = v1 \* v + v1;

cout << "Optimal speed: " << v3 << endl;

}

## Задание 3

/\*Три города нуждаются в мощных телевышках для улучшения качества телепередач.

Специалисты рассчитали, что можно обойтись одной вышкой,

если построить ее на одинаковом удалении от этих городов.

Составить программу определения расстояния от городов до вышки, если известны расстояния между городами.\*/

**void** **task3**(){

**int** A\_town = **10**; // город А

**int** B\_town = **15**; // город В

**int** C\_town = **20**; // город С

**double** p = (A\_town + B\_town + C\_town)/**2.0**;

**int** star = A\_town \* B\_town \* C\_town;

**double** back = p \* (p - A\_town) \* (p - B\_town) \* (p - C\_town);

**double** R = star / (sqrt(back) \* **4**);

cout << "Distance = " << R << endl;

}

## Задание 4

/\*Производительность труда при выполнении некоторой работы повысилась на 40%.

На сколько процентов сократилось время необходимое для выполнения этой работы.\*/

/\* Производительность труда (ПТ) = Количество продукции / время её производства,

следовательно, повышение ПТ является следствием уменьшения затрат времени на производство

того же количества продукции

\*/

**void** **task4**(){

**double** W; // ПТ

**double** Q = **100.0**; // продукция

**double** t = **10.0**; // время

**double** pr = **40.0**;

W = Q / t;

W = W + (W / **100** \* pr);

**double** t\_new = Q / W;

**double** t\_pr = t / **100.0**;

cout << "New work is time: - " << **100** - (t\_new / t\_pr) << "%" << endl;

}

## Дополнительное задание 1

/\*Сначала суток прошло n секунд, определить:

сколько полных часов прошло с начала суток,

сколько полных минут прошло с начала текущего часа,

сколько полных секунд прошло сначала текущей минуты. \*/

**void** **task\_addition1**(){

**int** sec = **10000**;

cout << "Full hours: " << (sec / **3600**) % **24** << endl;

cout << "Full minuts: " << (sec / **60**) % **60** << endl;

cout << "Full secunds: " << sec % **60** << endl;

}

## Дополнительное задание 2

/\*Три подружки договорились купить к праздничному столу 12 пирожных.

Первая купила 5 пирожных, вторая - 7, а третья вместо своей доли пирожных

внесла 120 тенге. Как подружкам разделить между собой эти деньги, чтобы всем потратить поровну? \*/

**void** **task\_addition2**(){

**int** p1 = **5**;

**int** p2 = **7**;

**int** p3 = **120**;

**int** full = **12**;

cout << "Friend 1: " << p1 \* (**120** / **12**) << endl;

cout << "Friend 2: " << p2 \* (**120** / **12**) << endl;

}

## Сложное задание

/\*Поезд проходит мост длиной 450 метров за 45 секунд, а мимо столба - за 15 секунд. Вычислите длину

поезда и его скорость.

Замечание:

- Фраза «поезд проходит мост за 45 с» означает, что 45 с - это время с момента, когда начало поезда вступит

на мост, до момента, когда конец поезда сойдет с моста.

- Фраза «мимо столба поезд проходит за 15 с» означает, что поезд свою длину мимо любой точки протягивает за 15 с.\*/

**void** **task\_hard1**(){

**int** most = **450**;

**int** time\_on\_most = **45**;

**int** time\_for\_stolb = **15**;

**double** speed = most / time\_on\_most;

**double** len = time\_for\_stolb \* speed;

cout << "Speed train = " << speed << endl;

cout << "Length = " << len << endl;

}