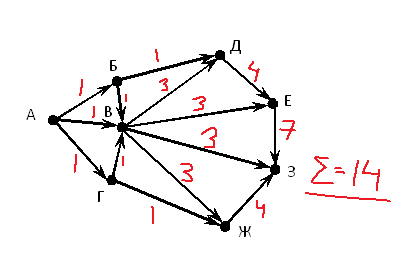
Инструкция к ЕГЭ (информатика)

По блоку 21-23

Есть дополнения б16 и б11

метод **«ПОЛЯНСКИХ»**



|  |  |
| --- | --- |
| Задание 21. **Разновидность 1.**  Опре­де­ли­те, какое число будет на­пе­ча­та­но в ре­зуль­та­те вы­пол­не­ния сле­ду­ю­ще­го ал­го­рит­ма:    **Var a,b,t,M,R :integer;**  **Function F(x:integer):integer;**  **begin**  **F:=(x+5)\*(x+3);**  **end;**  **BEGIN**  **a:= -5; b:=5;**  **M:=a; R:=F(a);**  **for t:=a to b do begin**  **if (F(t)> R)then begin**  **M:=t;**  **R:=F(t);**  **end;**  **end;**  **write(R);**  **END.** | Алгебраическая задача написанная на паскале, обращается внимание на знак “>”- находим наивысшую точку, если “<” наименьшую, в нашем случаем наивысшую.  И смотрим в конце  -если write (R) ответ по Игрику  -если write (M) ответ по Иксу  По формуле:  **F:=(x+5)\*(x+3);**  Есть диапазон **a:= -5; b:=5;**  **Это на оси x от -5 до 5**  **Находим точки Y по формуле y=( x+5)\*(x+3);**    Наибольшая точка лежит в координате (5;80)  Т.к write R – ответ по Y, это 80 |
| Задание 21 **Разновидность 2 (x)**  Опре­де­ли­те, какое число будет на­пе­ча­та­но в ре­зуль­та­те вы­пол­не­ния сле­ду­ю­ще­го ал­го­рит­ма:  Var a,b,t,M,R:integer;  Function F(x:integer):integer;  begin  F:= x\*x + 6\*x + 10;  end;  begin  a:=-10; b:=10;  M:=a; R:=F(a);  for t:=a to b do begin  if (F(t)> R)then begin  M:=t;  R:=F(t);  end;  end;  write(M);  end. | write(M);  **ответ по X**  **Знак больше, значит наиб точку**  F:= x\*x + 6\*x + 10; формула  a:=-10; b:=10; Диапазон по Х  **Находим точки Y**    Наивысшая точка лежит в координате (10;170)  Ответ по Х, значит 10 |
| Задание 21. **Разновидность 3 (редкая)**  Напишите в ответе число различных значений входной переменной k, при  которых программа выдаёт тот же ответ, что и при входном значении k = 64.  Значение k = 64 также включается в подсчёт различных значений k.  Для Вашего удобства программа приведена на пяти языках программирования. | Редкая задача, не сложная.  Дана функция  f:=n\*n  значит F(1) = 1\*1, т.к n в данном случае это 1  F(2) = 4  F(3)=9 и т.д  Они хотят, чтобы мы ввели число к, равное 64, прогнали это число и получили ответ это (i)  И вопрос такой, сколько еще чисел подошло бы помимо введенного к=64, чтобы i ответ был бы тот же.  Прогоняем к=64  i=12  k=64  условие пока i>0 и f(i)>=к выполняем i=i-1, т.е I уменьшается на 1 раз в каждом цикле.  12>0 и f(12)=144 >64  I=11  11>0 и f(11)=121 >64  ….  …  8>0 и f(8)=64 >=64  I=7 всё дальше не пойдет ответ 7,  7>0 и f(7)=49 >= 64(нет)  Сколько еще чисел бы подошли. Чтобы I был бы 7.  Видно, что там 49 чтобы дальше цикл не пошел можно взять от 50, значит, нам подошли бы числа от 50 до 64 включительно. В условии (Значение k = 64 также включается в подсчёт различных значений k)  Это 15 значений, ответ 15 |
| Задание 22. **Разновидность 1.**  Пе­ре­мен­ные http://reshuege.ru/formula/9d/9dd4e461268c8034f5c8564e155c67a6.png и http://reshuege.ru/formula/41/415290769594460e2e485922904f345d.png опи­са­ны в про­грам­ме как це­ло­чис­лен­ные. Опре­де­ли­те зна­че­ние пе­ре­мен­ной http://reshuege.ru/formula/9d/9dd4e461268c8034f5c8564e155c67a6.pngпосле вы­пол­не­ния сле­ду­ю­ще­го фраг­мен­та про­грам­мы:     |  |  | | --- | --- | | Бэй­сик | x = 432  y = x DIV 100  x = ( x MOD 100 ) \* 10  x = x + y | | Пас­каль | x : = 432;  y : = x div 100;  x : = ( x mod 100 ) \* 10;  x : = x + y; | | Ал­го­рит­ми­че­ский язык | x : = 432  y : = div (x, 100)  x : = mod (x, 100 ) \* 10  x : = x + y | | Просто вычисляем по алгоритму  Div поделить и показать цел часть  Mod поделить и показать остаток  X=432  Y=432 div 100 = 4.32 (остается 4)  X=(432 mod 100) \*10 = 32\*10=320 (когда его нашли тот первый Х исчез)  X=320+4= 324  Ответ 324 |
| Задание 22. **Разновидность 2.**  Опре­де­ли­те зна­че­ние пе­ре­мен­ной c после вы­пол­не­ния сле­ду­ю­ще­го фраг­мен­та про­грам­мы (за­пи­сан­но­го ниже на раз­ных язы­ках про­грам­ми­ро­ва­ния):     |  |  | | --- | --- | | Бэй­сик | Пас­каль | | a = 15  b = 30  b = a \* 2 - b / 2  If a > b Then      c := 3 \* b –a / 3  Else c = 3 \* a - 4 \* b  End If | a := 15;  b := 30;  b := a \* 2 - b / 2;  if a > b then      c := 3 \* b – a / 3  else      c := 3 \* a - 4 \* b; | | Здесь обращаем внимание на  if a > b then      c := 3 \* b – a / 3 (1условие положительное)  else      c := 3 \* a - 4 \* b; (2ое отрицат-е)  т.е вычисляем b, потом идет сравнение с а.  a=15  b=30  b=15\*2 - b/2=30 – 15 = 15 (помним что умножение и деление главнее) забываем про b=30  15>15 т.к неравенство неверно (идет 2ое условие отриц-е)      c := 3 \* a - 4 \* b = 45 – 60 = -15  Ответ -15 |
| Задание 22. **Разновидность 3.**  У ис­пол­ни­те­ля Каль­ку­ля­тор две ко­ман­ды:    **1. при­бавь 2**  **2. умножь на 3.**    Пер­вая из них уве­ли­чи­ва­ет число на экра­не на 2, вто­рая — утра­и­ва­ет его. Сколь­ко раз­лич­ных чисел можно по­лу­чить из числа 2 с по­мо­щью про­грам­мы, ко­то­рая со­дер­жит ровно 3 ко­ман­ды? | Даны 2 команды  +2  \*3  Вопрос, за 3 шага, с помощью этих команд сколько можно получить чисел? |
| Задание 22. **Разновидность 4.**  У ис­пол­ни­те­ля Каль­ку­ля­тор две ко­ман­ды, ко­то­рым при­сво­е­ны но­ме­ра:    **1. при­бавь 2,**  **2. умножь на 3.**    Пер­вая из них уве­ли­чи­ва­ет число на экра­не на 2, вто­рая — уве­ли­чи­ва­ет его в 3 раза.  Про­грам­ма для Утро­и­те­ля — это по­сле­до­ва­тель­ность ко­манд.  Сколь­ко есть про­грамм, ко­то­рые число 1 пре­об­ра­зу­ют в число 25?  Ответ обос­нуй­те. | Надо превратить 1 в 25. Расписать все варианты событий и подсчитать все варианты.  Когда умножаете, превышать нельзя, например 9\*3=27, нам надо ровно 25, значит от 9 до 25 идет 1 программа.    В конце считаем сумму программ. Их здесь 8. |
| Задание 23. **Разновидность1.**  Сколь­ко су­ще­ству­ет раз­лич­ных на­бо­ров зна­че­ний ло­ги­че­ских пе­ре­мен­ных x1, х2, хЗ, х4, х5, хб, х7, х8, ко­то­рые удо­вле­тво­ря­ют всем пе­ре­чис­лен­ным ниже усло­ви­ям?    (x1 —> х2) —> (хЗ—> х4) = 1  (хЗ —> х4) —> (х5 —> хб) = 1  (х5 —> хб) —> (х7 —> х8) = 1    В от­ве­те не нужно пе­ре­чис­лять все раз­лич­ные на­бо­ры зна­че­ний пе­ре­мен­ных x1, х2, хЗ, х4, х5, хб, х7, х | To be announced. |
| Задание 23. **Разновидность 2.**  Сколь­ко раз­лич­ных ре­ше­ний имеет урав­не­ние    (K ∨ L) ∧ (M ∨ N) = 1    где K, L, M, N – ло­ги­че­ские пе­ре­мен­ные? В от­ве­те не нужно пе­ре­чис­лять все раз­лич­ные на­бо­ры зна­че­ний K, L, M и N, при ко­то­рых вы­пол­не­но дан­ное ра­вен­ство. В ка­че­стве от­ве­та вам нужно ука­зать толь­ко ко­ли­че­ство таких на­бо­ров. | Здесь нужно указать число наборов (комбо)    Сумма равно 9 (9 комбинаций) |
| Задание 23. **Разновидность 3.**  Ка­ко­во наи­боль­шее целое число X, при ко­то­ром ис­тин­но вы­ска­зы­ва­ние    (50 < X·X) → (50 > (X+1)·(X+1))? | Упрощаем  (50 < X·X) → (50 > (X+1)·(X+1))    Здесь уже видно, что это число 7, в первой скобке получим истину, т.к 50>49. Во второй уже будет ложь т.к 50>64(не работает).  7 является наибольшим, т.к если возьмем 8 и более в обеих скобках будет ложь. |
| Дополнительно Б11  **Разновидность 1.**  Ниже на пяти язы­ках про­грам­ми­ро­ва­ния за­пи­сан ре­кур­сив­ный ал­го­ритм F.     |  |  | | --- | --- | | **Бей­сик** | **Python** | | SUB F(n)  **PRINT** n  **IF** n < 5 **THEN**          F(n + 1)          F(n + 3)  **END** **IF**  **END** SUB | def F(n):      print(n)      if n < 5:          F(n + 1)          F(n + 3) | | **Пас­каль** | **Ал­го­рит­ми­че­ский язык** | | **procedure** F(n: **integer**);  **begin**  **writeln**(n);  **if** n < 5 **then**  **begin**        F(n + 1);        F(n + 3)  **End;**  **end** | **алг** F(**цел** n)  **нач**  **вывод** n, нс  **если** n < 5 **то**      F(n + 1)      F(n + 3)  **все**  **кон** |   Чему равна сумма всех чисел, на­пе­ча­тан­ных на экра­не при вы­пол­не­нии вы­зо­ва F(1)? | Смотрим на паскаль:  procedure F(n: integer);      begin    writeln(n);    if n < 5 then      begin        F(n + 1);        F(n + 3)      End;      End  Даны функции F(n), т.е F(1) F(2) и т.д.  n<5 значит формулы  F(n + 1);        F(n + 3) работают для F(4), F(3) и ниже  А другие которые 5 и больше равны своим n  F(5)=5  F(6)=6  F(7)=7 и т.д  Общая формула для условия (n<5) следующая  F(n)=n+F(n+1)+F(n+3)  По условию надо найти F(1)  F1=1+f2+f4  F2=2+f3+f5  F3=3+f4+f6  F4=4+f5+f7  F7=7  F6=6  F5=5  F4=4+f5+f7=4+5+7=16  F3=3+f4+f6=3+16+6=25  F2=2+f3+f5=2+25+5=32  F1=1+f2+f4=1+32+16=49  Ответ 49 |
| Дополнительно Б11  **Разновидность 2. Та же задача.**  Ниже на пяти язы­ках про­грам­ми­ро­ва­ния за­пи­сан ре­кур­сив­ный ал­го­ритм F.     |  |  | | --- | --- | | Бей­сик | Python | | SUB F(n)      IF n > 2 THEN          F = F(n - 1) +F(n-2)      ELSE          F = 1      END IF  END SUB | def F(n):      if n > 2:          return F(n-1)+ F(n-2)      else: return 1 | | Пас­каль | Ал­го­рит­ми­че­ский язык | | procedure F(n: integer): integer;  begin      if n > 2 then      F := F(n - 1) + F(n - 2)      else          F := 1;  end; | алг цел F(цел n)  нач  если n > 2  то      знач := F(n - 1)+F(n - 2)  иначе      знач := 1  все  кон |     Чему будет равно зна­че­ние, вы­чис­лен­ное ал­го­рит­мом при вы­пол­не­нии вы­зо­ва F(5)? | procedure F(n: integer): integer;  begin      if n > 2 then      F := F(n - 1) + F(n - 2)      else          F := 1;  end;  Если n>2  Значит  F(3), F(4), F(5) и т.д имеют общую формулу  F := F(n - 1) + F(n - 2)  Else т.е у которых n>2  F(2), F(1), F(-1), F(-2) и т.д равны 1 (F := 1;)    По условию ищем F(5) ???  F5=F4+F3  F4=F3+f2  F3=F2+F1=2  F4=2+1=3  F5=3+2=5  Ответ: 5 |
| Дополнение. **Задание Б16. (редкое)**  Сколь­ко еди­ниц со­дер­жит­ся в дво­ич­ной за­пи­си зна­че­ния вы­ра­же­ния: 42020 + 22017 – 15? | В такой задаче сначала нужно привести к общей системе счисления, именно к двоичной, т.к считаем кол-во единиц. |

