Конспект лекции 1

Различия между языками программирования. Люди часто думаю что язык програмирования который они учат это одно и тоже что и язык программирования. Есть явление Ява головного мозга – мало где пишут так же как пишут на Ява но программисту сложно перестроиться если он уходит в другой язык не дай боже.

Если мелкому ребенку дать молоток то у него все вокруг превратится в говзди – это пример ограниченности понимания своего инстурмента и узколобости. В общем у каждого ЯП есть свои особенности и зачастую не надо стремтиться переносить особоенности из одного ЯП в другой.

Есть смысл а есть средство донесения смысла. Язык это средство донесения смысла, а например объект собака он останется тем же на любом языке.

Непонятно зачем мне это рассказывают про радугу, лягух и собак. Немецкий язык хорошо подходид для юристов – все однозначно. Русский язык – отлично подходит для поэзии. Английский язык лучше всего подходит для математики и для программирования.

Значки в математике называются кванторы – это смысловые значки. Нотная грамота это тоже язык записи информации.

Очень много воды, вроде хорошо расскаывает, но зачем столько простой инфы, будто бы мы совсем человеки неразумные.

ЗАДАЧА1

Есть 5 гриек, все они разного веса. Надо найти вес самой тяжелой гири. Что можно делать:

1. Взять гирю 2) сравнить по весу 3) поменять местами 4) запомнить вес гири

Нужно расписать алгоритм из этих действий для сортировки.

Дали 5 мин.

1. Взять 1ю гирю
2. Сравнить ее со 2й
3. Если 1я тяжелее второй то запомнить ее вес
4. Если 2я тяжелее й1 тозапонмить вес 2й гири. Допустим запомнили вес 2й гири.
5. Взять 3ю гирю.
6. Сравнить со 2й. Если 2я тяжеелее 3й то запомнить вес 2й. Допустим 2я тяжелее.
7. Взять 4ю гирю.
8. Сравнить вес со 2й.
9. Если 4я тяжелее то запонмить ее вес. Допустим запомнили 4ю.
10. Взять 5ю гирю сравнить ее вес с 4й. Если 4я тяжеелее то запонмить ее вес. Допустим 4я тяжелее.
11. Нашли ответ: 4я гиря самая тяжелая.

Чтобы не записывать так словами алгоритмы люди придумали графическое изображение алгоримтов, чтобы его изображение было универсальным в планетарном масштабе и чтобы не зависить от языка разговора человека. Блоксхема по сути это тоже язык, но он достаточно прост для изучения.

Препод говорит что якобы ПК не понимает блоксхемы, но на самом деле это не так. Это все зависит от IDE, например Flowgorithm понимает живые алгоритмы и даже умеет показывать прграммный код сгенерированный из блоксхемы. ВОбщем, это вопрос комплиятора. Но я понимаю, что все серьезные разработки ведутся на текстовом языке а не визуальном.

Лабиринт.

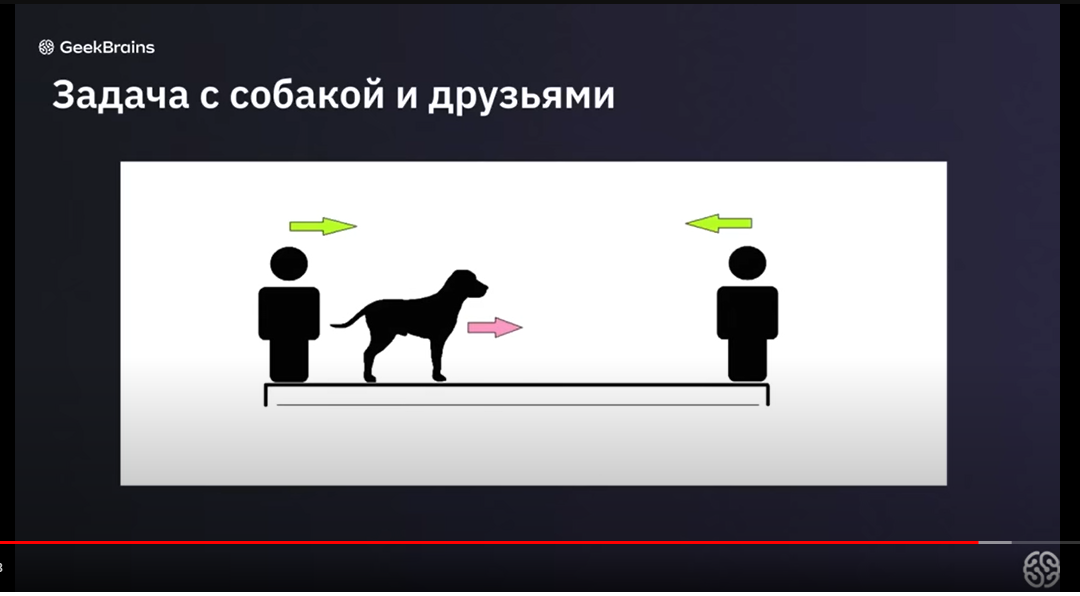
Ищем выход из него. Записываем алгоритм для выхода из лабиринта. Правило левоц/правой руки. Держимся вобщем одной рукой за стену и ползем ею по стене пока идем и не отрываем и так га каждом повороте и в итоге выйдем.

ПРИМЕРНЫЙ АЛГОРИТМ:

1. Если левая рука касается стены то идем вперед и держимся за стену.
2. А если рука не касается то надо повернуться подойти к стене и положиуть руку на стену.
3. Если впереди стенка то повернуть направо.
4. Если впереди свободно то идем вперед.
5. Мы уже вышли из лабиринта? Если да то алгоритм завершен. Если нет то идем дальше.
6. И сначала с шага1 повторим.

ЗАДАЧА С ОДНЙО СОБАКОЙ и ДВУМЯ ДРУЗЬЯМИ

Два друга движутся друг другу с разной скоростью. Собака знает обоих друзей, она бегает от одного к другому со своей скоростью потом возвращается обратно и потом снова обратно.



Вопрос: сколько раз собака пробежала дорогу туда-обратно между друзьями до их столкновения носами.

АЛГОРИТМ

1. Дано: СкоростьДруга1 = v1
2. Дано СкоростьДруга2 = v2
3. Дано СкоростьСобаки = vdog
4. Дано НачальноеРасстояниеМеждуДрузьями = Space
5. Кол-воХодокСобаки=0

Рассуждения об алгритме:

Собака стартует с точки x1 и пробегает до второго друга расстояние s1 с постоянной скоростью vdog. Зная время затраченное собакой на первую ходку от Друга1 до Друга2, вычислим из него какое расстояние за это время прошел Друг1 со СкоростьюДруга1 и какое расстояние прошел Друг2 со СкоростьюДруга2 Расстояние1+Расстояние2=Дистанция[1]. Затем вычтем эту сумму расстояний двух друзей из : Space=Space-Дистанция[1]. Если Space>0 то друзья все еще не дошли друг для друга. Теперь инкрементируем Кол-воХодокСобаки++.

Вышло 5 минут. Препод рассказывает про стиль нименования CamelCase , в JAVA это основной стиль. А вот в python оснвным является SnackeCase.

Кстати, нужно задавать именованные переменные при решении задачи , это дает нам:

1. Мы понимаем что конкретная переменная представляет из себя.
2. Мы можем изменить значение числа при первой инцииализции переменной и потом она поменяется во всей рпогармме.

Сомневаюсь что кто то будет хоть раз читать мой конспект среди десятков конспектов особенно из числа студентов. Разве что искусственный интеллект проверить его на цензуру перед выкладкой в общую кучу.

Только что увидел что лектор Ильнар Шафигуллин это идеолого программы «Разработчик» . Прикольно.