

Computational Thinking Approach (Bilişimsel Düşünme Yöntemi)

problem çözmeyi, eleştirel düşünmeyi ve insan-bilgisayar etkileşimi arasındaki ilişkiyi anlamayı hedefleyen çalışmaların bütünü

pseudo code algorithm (Sözde Kod Algoritması)

Bir problemin çözümünü verecek olan algoritmanın, programlama dilinin yazım kurallarından (syntax) bağımsız, kendi doğal dilimizde tasarlanmasıdır. Bu tasarımı yaparken bilişimsel düşünme yöntemini (computational thinking approach) kullanırız.

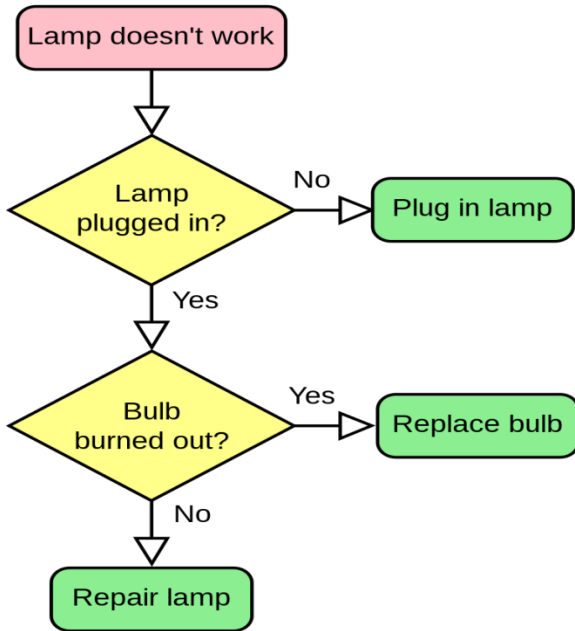
Örn:

kullanıcıdan yaş girmesini isteyip. Girilen yaş 18 den büyükse Reşitsiniz, değil ise reşit değilsiniz yazdıran algoritma

- 1.başla
2. Oku “ yaşıınız ?” (Y)
3. Eğer
 - 3.1 (Y >=18) yaz “Reşitsiniz”
 - 3.2(DEĞİLSE) yaz “Reşit değilsiniz”
4. Dur

Flow Chart (Akış Şeması)

Problemin şekillerle ifade edilmesidir.



Workshop #1

Ödevi Zamanında Verememe/Eksik Ödev Teslimi

Bir Paket 10'lu kahve getirilecek

Giriş

Kodlar Visual Studio Code editörü ile yazılacak. Bu editörde yer alan konsolda kaynak kodunuz **gcc** derleyicisi ile derlenecek.

Derlenen programa prg_WxPy ismi verilecek(x workshop ve y part numarası).

Bütün workshop çalışmaları kendi numaraları ile Workshops klasörünüzde arşivlenecek

Konular

- Standart çıktı kullanımını geliştiren basit bir c programı yazabilme
- Visual Studio Code tümleşik entgre geliştirme ortamını IDE (Integrated Development Environment) tanıma ve kullanma

Beklenti

- Visual Studio Code IDE ile bir c dosyası oluşturuyor olmak
- Visual Studio Code Konsolundan yazılan programı derleyebilmek
- Derleyici parametrelerinden –Wall ve –o parametrelerini tanımlayabilmek

Part-1 (%10)

Standart çıktıda aşağıdaki gibi çıktı veren bir c programı yazınız.

Workshop 1 Part-1

=====

I'm displaying this using the 'printf' stdio
(standard input output) library function!

Dear teacher,

I promise I will work hard from this day onward.
I acknowledge that practice is extremely important,
so I will do all workshops, quizzes, and assignments.

Sincerely,

First_Name Last_Name
Student ID# 123456789

Not

- Sarı alanlara değişkenlerde yer alan kişisel bilgiler yerleştirilecek.
- Elbette çıktı sarı renkli görünmeyecek
- Kaynak kodunuzun adı w1p1.c olacak
- Oluşacak dosyanın adı prg_w1p1 olacak
- Gcc ile -Wall parametresi mutlaka kullanılacak

Part-2 (%40)

Bu kısım önceki kısmın devamı olacak. Vsc editörünüz ile ws1 klasörünüzü proje klasörü yapın.

Proje klasörünüze w1p2.c dosyası ekleyin. “w1p1.c” dosya içeriğini “w1p2.c” dosyasına yapıştırın.

“w1p2.c” dosyasını aşağıdaki gibi modifiye edin.

Workshop 1 Part-2

=====

Using tab specifiers...

<t> Tab-1<t>Tab-2<t>Tab-3

<t> -----<t>====<t>-----

\ this is a back-slash character!

% this is a percent sign character!

" this is a double-quote character!

My favourite quotes are:

1.<t> "Skill is only developed by hours and hours of work."

<6t> -Usain Bolt

2.<t> "It's not about having time. It's about making time."

<6t> -unknown

3. <t> "All of us do not have equal talent. But, all of us
<t> have an equal opportunity to develop our talents."

<6t> -A.P.J. Abdul Kalam

Not

- <t> , ascii tablosundaki tab karakterinden bir adet
- <6t> , ascii tablosundaki tab karakterinden 6 adet
- Warning çıktılarının görünmesini sağlayacak şekilde program derlenecek
- Derlenen dosya adı **prg_w1p2** olacak