

Ödev 3 - Sanal Fabrikam

Bir fabrikada makinelerin çalışmasını planlayacak bir program yazacaksınız. Program bulunduğu klasörden üç metin dosyası okuyacak ve zamanlama sonucunuz için bir çıktı dosyası oluşturacaksınız. Tüm süreler dakika cinsinden ve tüm boyutlar metre cinsindendir.

Aşağıda her dosyanın açıklamasını ve biçimini bulabilirsiniz;

Operations.txt: Bu dosya, işlemlerin makine tarafından yapılabileceğini gösterir. Bu dosya, her satırda iki bilgi içerir. Birincisi işlem kodu, ikincisi ise hızıdır (dakikada iş yapılabilir).

Biçim: (İşlemKodu; Miktar \ n \ r)

1; 10 2; 15 3; 100 4; 50

İlk satır, bu makinenin operasyon kodu 1'i yapabileceği ve hızı dakika başına 10m olduğu anlamına gelir.

SetupDuration.txt: Bu dosya, işlem değişiklikleri için kurulum sürelerini gösterir. Başlangıç işleminin sıfır kurulum süresi vardır. Bununla birlikte, makineyi bir işlemten diğerine geçirdiğinizde, ikinci işlemi başlatmak için bu kurulum süresini alır. Her işlem çiftinin bu dosyada bir kaydı olduğu için kurulum süreleri iki yönden eşittir.

Biçim: (İşlemKodu1; İşlemKodu2; KurulumSüresi \ n \ r)

1; 2; 10
1; 3; 10
1; 4; 12
2; 3; 5
2; 4; 3
3; 4; 20

İlk satır, işlem1'den işlem2'ye geçerseniz makineyi hazırlamanın 10 dakika sürdüğü anlamına gelir.

Orders.txt: Bu dosya, müşterilerin makinelerimizde planlanması gereken siparişlerini içerir. Tüm işleri tek tek basitleştirebiliriz ancak kurulum sürelerini azaltarak makine performansını artırmak istiyoruz. Öte yandan, her siparişin bitmesi gereken bir son tarihi vardır. Bu nedenle son teslim tarihlerine uymalı ve makinenin performansını artırmalıyız. Bu dosya bize dakika cinsinden sipariş kodunu, metre cinsinden sipariş miktarını, işlem türünü ve son teslim tarihini verir.

Bişim: (SiparişKodu;İşinUzunluğu;İşlemKodu; SonTeslimTarihi \ n \ r)

1; 100; 2; 300

2; 100; 3; 300

3; 200; 1; 500

4; 200; 1; 500

5; 100; 2; 1000

6; 100; 2; 1000

7; 100; 4; 1000

Dördüncü sıra, Sipariş 4'ün ilk işlem türü ile yapılması gereken 200 metre olduğu ve bu sipariş için son tarih şu andan itibaren 500 dakika olduğu anlamına gelir. Bu nedenle, bu siparişi başlama saatinden 301 dakika önce planlamanız gerekir (Kurulum sürelerinin farkında olun).

Bir işin Süresi bir kayan noktalı sayı ile sonuçlanırsa, bunu en yakın üst tam sayı olarak almanız gerekir. (Örn: Dakikada 4m'lik bir işlem için, 22m'lik bir iş 5,5 dakikada yapılabilir, ancak bunu 6 dakika olarak almanız gerekir.) Dosyalardaki tüm verilerin tamsayı olması gerektiğinden.

Girdi dosyaları belleğinizin izin verdiği kadar olabilir. Programınız tek bir dosyada 1.000.000 siparişi işleyebilir.

PROGRAMIN ÇIKTISI

Programınız zamanlama çıktısını bir metin türü dosyası olarak oluşturacak ve programın toplam süresini ekranda gösterecektir. Çıktı dosyası için açıklama aşağıdadır;

Schedule.txt: Program sonucunuzu, girdi dosyaları formatı gibi içerir. Her satırda zamanlama zamanını, İşlem Kodunu, Sipariş Kodunu, Sipariş Miktarını ve Kurulum Ek yükünü göstereceksiniz.

Biçim: (ScheduleTime; OperationCode; OrderCode; AmountOfWork; SetupOverhead \n \r)

Ör:

0; 2; 1; 100; 0

17; 3; 2; 100; 10

Bu iki satır, ilk sırayı ilk sırada planladığımız anlamına gelir. T = 0'dan başlar ve 2. operasyon tipinde 100m çalışması vardır. İlki için kurulum maliyetimiz yoktur.

İkinci planlanan sipariş, ikincisidir. T = 17'de başlar çünkü önceki iş için makinenin hızı dakikada 15m'dir, bu nedenle 100 m'lik iş 6,6 dakikada yapılabilir. Bunun yerine 7 dakika ve 2 ila 3 işlem arasında kurulum süresi için 10 dakika sürüyoruz. İkinci dereceden hesaplanan başlangıç zamanı 10 + 7: 17 olarak sonuçlanır.

ÖDEV TESLİMİ HAKKINDA ÖNEMLİ BİLGİ

Örnek girdi ve çıktı dosyaları ekte bulunmaktadır.

Ödevler sitemizde sistem tarafından toplanmaktadır. Başka herhangi bir teslimat yöntemi kabul edilmeyecektir. Kodu yazarken öğrenciler aşağıdaki kurallara uymalıdır.

- Kod dosyanızın başında, sizin hakkınızda bilgi veren bir yorum bölümü bulunmalı, kodunuzun derleyici ortamı ve kod dosyasını çalıştırmak için diğer her türlü konu dikkate alınmalıdır.

Ör:

```
// Özgür Can TURNA
```

```
// 1306xxxxxx
```

```
// Tarih: 28.04.2016
```

```
// Geliştirme Ortamı: Visual Studio2019
```

- Kod dosyası Microsoft Visual Studio'da veya GCC altında derlenmelidir (CodeBlocks bunu da kullanıyor)
- Yalnızca bir kod dosyası toplanır, bu nedenle başka bir dosya veya birden fazla dosya göndermeye çalışmayın.
- Kod dosyası öğrenci numaranız "1306000001.cpp" olarak adlandırılmalıdır
- Programınız "stdafx.h" gibi fazladan başlık dosyaları içeriyorsa, kodunuzu derleyemediğim için sıfır alır. Dikkatli olun.