**Operating Systems – Homework 1 *Mert Balkan - 20253508***

fork(): Yeni bir işlem oluşturmak için yazılır.

fork() fonksiyonu çağırıldıktan sonra, programda yürütülen işlemden yeni bir çatal oluştur ve ***child process*** olmuş olur. Yani fork çağırıldıktan sonra programda hem child hem parent processler aynı anda çalışmaya devam eder ve bundan sonraki kod bloklarını kendi içlerinde çalıştırırlar. Eğer pid değişkeninin değeri **0’dan büyükse** o halde çalışan process ***parent processdir***, şayet **değer 0’a eşitse o halde çalışan process *child processdir***. “pid” değişkeninin değeri her ikisine de eşit değilse o halde bir hata oluşmuştur.

Aşağıda ise verilen ödevin kodu vardır.

**metin içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**Kodun çıktısı:**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Kodun çıktısından da görüleceği üzere, parent ve child processlerin bitme veya başlama zamanları birbirinden farklı olabilir. Kimi yerde bazı parent processler daha erken biterken, kimi yerde bazı child processler daha erken bitebilir. Kodu her defasında en baştan derleyip çalıştırırsak bu farkı çok daha iyi görebiliriz.

Ve yine kodun çıktısında da gördüğümüz gibi, bazı processlerin parent id’leri 1, yani processlerin çalışan parent processleri, kendilerinden daha önce bitmiş ve bu processler yetim kalmış. Bu wait(NULL) fonksiyonu çağırılarak, önce hangi processlerin çalışması gerektiğine karar verilerek önlenebilirdi.