



**BURSA TEKNİK
ÜNİVERSİTESİ**

MÜHENDİSLİK VE DOĞA BİLİMLERİ FAKÜLTESİ

Bilgisayar Mühendisliği Bölümü

**ALGORİTMALAR VE PROGRAMLAMA DERSİ
PROJE ÖDEVİ RAPORU**

OKAY BÜYÜKDEVECİ

21360859070

ESRA ÖZCAN

21360859034

BLM0111 SUBE 2

ARALIK 2023

1-GİRİŞ

Projemiz bir sanal köpek yetiştirme oyunu olarak tasarlandı. Tanıdığımız bir köpek dostumuzdan esinlenerek Talya ismini seçtik. Kod çalıştırıldığında karşımıza karşılama metni ve köpeğimizle ilgili bazı bilgiler geliyor. Projemizde Talya adındaki köpeğimizin bazı ihtiyaçları bulunmakta ve kullanıcıdan 28 aktivite bulunan menüden herhangi bir aktivite seçmesi bekleniyor. Seçilen aktiviteye göre ihtiyaç düzeylerinde belirli artışlar ve azalışlar meydana geliyor. Her defasında düzeyleri ve menüyü ekrana getiriyoruz, eğer kullanıcı çıkış yapmak isterse menümüzün son seçeneğinde çıkış bulunmakta, 28. Seçeneği seçerek çıkış yapılabilir. Onun haricinde köpeğimizin belirli ihtiyaçlarının düzeylerinin fazla artışı ya da azalışı yaşamını yitirmesine sebep olabiliyor.

//kim ne yaptı

2-İHTİYAÇLAR

Köpeğimizin sahip olduğu bazı temel ve ek ihtiyaçlar mevcut. Bunlar; tokluk, uyku, sevgi, sosyallik, sağlık, eğlence, hijyen, tuvalet, eğitim, huzur olarak projeye eklendi. Bu ihtiyaçların her biri kendi içinde 3 ayrı fonksiyona ayrılıyor.

Tokluk ihtiyacı için yazdığımız fonksiyonlar *yemekYe()*, *suIc()*, *yeniLezzetlerDenemek()*; **sevgi** ihtiyacı için yazdığımız fonksiyonlar *sahibiTarafındanSevilmek()*, *yabanciBiriTarafındanSevilmek()*, *karsiCinsTarafındanSevilmek()*; **uyku** ihtiyacı için yazdığımız fonksiyonlar *uykuVakti()*, *zorlaUyandır()*, *geziVakti()*; **sosyallik** için yazdığımız fonksiyonlar *kopekParkındaVakitGecirmek()*, *evdeDurmak()*, *asikOlmak()*; **sağlık** için yazdığımız fonksiyonlar *veterinereGitme()*, *hastaOlma()*, *genclikIksiri()*; **eğlence** için yazdığımız fonksiyonlar *sahibiIleVakitGecirme()*, *digerHayvanlarlaVakitGecirme()*, *denizdeYuzme()*; **hijyen** için yazdığımız fonksiyonlar *dusAlmak()*, *kişiselBakim()*, *camurdaOynamak()*; **tuvalet** için yazdığımız fonksiyonlar *ihitiyacYuruyusu()*, *tuvaleteCikma()*, *tuvaletEgitimi()*; **eğitim** için yazdığımız fonksiyonlar *hocaylaEgitimeGitme()*, *rehberlikEgitimi()*, *korumaEgitimi()*; bütün bu fonksiyonların altında *cikis()* fonksiyonu tanımladık.

Bu fonksiyonları kullanıcının konsoldan yaptığı seçime göre çağırıyoruz ve her fonksiyonun etkilediği ihtiyaçların düzeylerini ekrana yazdırıyoruz. Örneğin; kullanıcı köpeğimiz Talya'nın sahibi tarafından sevilmesini istediğinde sevgi ihtiyacımızın düzeyi artıyor. Bu işlemten sonra ekrana aktivite sonuçlanma metni ve yeni düzeyler yazılıyor. Ardından kullanıcıdan yeni işlem yapması için ekrana tekrardan menü getiriliyor. Kullanıcı oyunu sonlandırmak isterse 28.seçenekle çıkış yapıyor ve oyun sonlanıyor.

2.1-Sevgi ihtiyacı

Köpeğimiz Talya her canlının ihtiyacı olduğu gibi sevgiye ihtiyaç duyabilir. Kötü bir dönemde olabilir bu nedenle sahibi tarafından, sevgilisi tarafından ya da yabancı biri tarafından sevilme isteyebilir. Eğer Talya'nın sevgiye ihtiyacı olduğunu düşünüyorsanız aşağıdaki fonksiyonlardan herhangi birini menüdeki fonksiyonla aynı isimli seçeneklerle seçebilirsiniz.

2.1.1-sahibiTarafındanSevilmek()

sahibiTarafındanSevilmek() fonksiyonu, Talya'nın sahibi tarafından sevildiği bir durumu modifiye eder. Bu fonksiyon çağrıldığında, genel olarak Sevgi, Eğlence ve Huzur artar, diğer ihtiyaçlar bir birim azalır.

2.1.2-yabanciBiriTarafındanSevilmek()

yabanciBiriTarafındanSevilmek() fonksiyonu, Talya'nın sokakta hiç tanımadığı biri ya da evlerine gelen ama daha önce hiç görmediği biri tarafından sevilmesini temsil eder. Bu fonksiyon çağrıldığında, genel olarak Sevgi, Eğlence ve Huzur artar, diğer ihtiyaçlar bir birim azalır.

2.1.3-karsiCinsTarafındanSevilmek()

karsiCinsTarafındanSevilmek() fonksiyonu Talya'nın hoşlandığı köpek Buddy tarafından ya da başka karşı cins köpekler tarafından sevildiğini, ilgi gördüğünü simgeliyor, bu durum Talya'nın sevgi düzeyini diğer aksiyonlara göre epey artırıyor. Bu fonksiyon çağrıldığında, genel olarak Sevgi, Eğlence, Sosyallik ve Huzur artar, diğer ihtiyaçlar bir birim azalır.

Bu fonksiyonları çağırırsanız Talya'nın sevgi, eğlence ve huzur düzeylerini artırmış olursunuz.

Sevilmenin ne kadar güzel etkileri varmış. Sizde Talya'nın sevilmesini istersiniz değil mi?

2.2-Tokluk ihtiyacı

Talya'nın her canlının yaşaması için ihtiyacı olduğu gibi su ve besine ihtiyacı var. Eğer besin ve suyu yeteri kadar almazsa hayati fonksiyonlarını yitirebilir. Talya'nın aç ve susuz olduğunu hissettiğiniz her an, menüdeki aşağıdaki fonksiyonlarla aynı isme sahip seçenekleri seçerek tokluk düzeyini artırabilirsiniz:

2.2.1-yemekYemek()

Bu fonksiyon, köpeğin yemek yemesini simüle eder. Kullanıcıya bir menü sunar ve seçilen yiyeceğe göre köpeğin ihtiyaç düzeylerini günceller. Ayrıca, köpeğin genel ihtiyaçlarından bazılarını azaltır.

Seçilen adet kadar kemik eklenir. Eğlence, tuvalet, huzur 2 arttırılır. menüden seçilen yiyeceğe göre farklı aksiyonlar alınır ve ardından genel ihtiyaçlar güncellenir. Örneğin, köpek kemiği zevkle yerken huzuru artar, ancak tuvalet ihtiyacı da artar.

2.2.2-sulcmek()

Bu fonksiyon, köpeğin su içmesini simüle eder. Su içme, genel ihtiyaçlarına pozitif etki eder.Tokluk 1 arttırılır, tuvalet 3 arttırılır, huzur 1 arttırılır. Genel ihtiyaçlar güncellenirken, diğer ihtiyaçlar bir birim azaltılır.

2.2.3-yeniLezzetlerDenemek()

Bu fonksiyon, köpeğin yeni lezzetleri denemesini simüle eder. Kullanıcıya bir lezzet seçimi sunar ve seçilen lezzetin köpeğin ihtiyaçlarına olan etkilerini günceller.

Örneğin ananas yerse tokluk 3, huzur 1 arttırılır, Uzak Doğu Yöresel Köpek Yemeği yerse tokluk 1, huzur 2 azaltılır. Sonrasında diğer ihtiyaçlar 1 birim azaltılır ve genel ihtiyaçlar güncellenir.

Talya daha önce yemediği mamaları tatmayı çok sever (bazen damak tadına uymasa da).Bu deneyimleri ona yaşatmak isterseniz de tokluk ihtiyacının 3.fonksiyonu işinizi görecektir. Denemek istediği yemekler de seçenek olarak kullanıcıya sunuluyor ama her deneyim iyi sonuçlanmayabilir değil mi?

Bu fonksiyonlar, köpeğin çeşitli aktivitelerle ve beslenme şekilleriyle ilişkilendirilmiş ihtiyaç düzeylerini güncelleyerek, köpeğin sanal bir simülasyonunu gerçekleştirmektedir.

2.3-Sosyallik ihtiyacı

Sevimli hayvan dostumuz Talya oldukça sosyal bir köpek. Çokça arkadaşı var. Talya'nın sosyallik ihtiyacını gidermek için kullanıcıya sunduğumuz 3 fonksiyonumuz bulunuyor:

2.3.1-kopekParkindaVakitGecirmek()

Bu fonksiyon, köpeğin bir köpek parkına gitmesini simüle eder Talya'ya 'Köpek parkında vakit geçirmek' aktivitesi yaptırılmak istendiğinde hangi arkadaşıyla vakit geçireceği kullanıcı tarafından seçiliyor, sevmediği arkadaş seçildiği takdirde 'Emin misin?' diye soruluyor sonucuna bağlı olarak aktivite metni ekrana yazdırılıyor. Eğer arkadaş seçiminde hoşlandığı köpek denk gelirse sevgi düzeyi normalden fazla artırılıyor.

Seçilen arkadaş göre köpeğin ihtiyaç düzeyleri üzerinde çeşitli etkiler yapar.

Arkadaşı COCO veya BUDDY ile vakit geçirmek, sevgi düzeyini ve sosyal aktivitesini artırır, eğlencesini artırır ve huzurunu güçlendirir, bunun yanı sıra yorulduğu için uykusunu getirir.

Arkadaşı BELLA ile vakit geçirmek, psikolojik durumu, sosyal aktivitesini artırır, eğlencesini artırır fakat diğer arkadaşlarının aksine huzur düzeyinde artış gözlenmez.

Genel ihtiyaçlar; tuvalet ve sağlık birer birim azalır.

2.3.2-evdeDurmak()

Bu fonksiyon, köpeğin evde vakit geçirmesini simüle eder. Bu durum, genellikle aktivitelerin sınırlı olduğu ve sıkıcı olduğu bir durumu temsil eder. Talya'ya 'Evde durma' aktivitesi yapıldığında sosyallik seviyesini azaltmış biraz da huzursuzlandırmış oluyoruz.

Sosyallik düzeyi üç birim azalır, eğlence seviyesi 2 birim azalır, huzur düzeyi 4 birim azalır.

Genel İhtiyaçlar: Tuvalet ve hijyen birer birim azalır.

2.3.3-asikOlmak()

Bu fonksiyon, köpeğin aşık olma durumunu simüle eder. Kullanıcıya köpeğin birine aşık olup olmadığını sormaktadır. Eğer Talya aşık olursa bu konu hakkında sahibine bir şeyler söylemek isteyip istemediğini soruyoruz. Sonuca bağlı olarak Talya aşık olduysa sevgi ve huzur seviyesinde artış; aşık olmadıysa sevgi ve huzur seviyesinde azalış, depresyon, tatlı yeme isteği ve tokluk seviyesinde artış meydana geliyor.

Bu fonksiyon, köpeğin duygusal durumunu değiştirerek ve genel ihtiyaçlarına etki ederek köpeğin simülasyonunu zenginleştirir.

2.4-Uyku ihtiyacı

Talya çok uykucu bir köpek, hemen her yerde uyuyabilir. Eğer kaybolduysa ve bulamıyorsanız bir yere saklanıp şekerleme yapıyor demektir. Talya'nın uyku ihtiyacını gidermek sık sık yaptırmanız gereken eylemlerden biri çünkü hep uykusu var!! Uyku ihtiyacı için de aynı şekilde 3 fonksiyonumuz var:

2.4.1-uykuVakti()

Bu fonksiyon, köpeğin uyuma eylemini simüle eder. Eğer Talya'nın uykuya ihtiyacı olduğunu düşünüyorsanız uyku vakti seçeneğini seçerek Talya'ya güzel bir uyku çektirebilirsiniz. Eğer eğlence düzeyi 10'dan küçükse, köpek uyur ve bazı ihtiyaçları değişir.

Öncelikle Talya'nın uyku düzeyi kontrol edilir, 10'dan küçükse Talya'nın uyku vaktidir. Eğlence düzeyi 10'dan küçükse, köpeğin uyuma zamanıdır. Eğlence 3 birim artar, sağlık 2 birim azalır, hijyen 2 birim artar. Bu fonksiyon diğer tüm fonksiyonların aksine özyinelemeli bir fonksiyondur, uyku düzeyi 10'a ulaşınca kadar fonksiyon kendini tekrar eder.

2.4.2-zorlaUyandır()

Bu fonksiyon, köpeğin zorla uyandırılma eylemini simüle eder. Kullanıcıya seçenekler sunar ve seçilen seçeneklere göre köpeğin durumu değişir. Bazen Talya çok garip yerlerde uyuya kalıyor, bunun için zorla uyandır seçeneğini seçebilir ve 'sahibinin yanına', 'kulübesine' ya da 'şöminenin önüne' yatırabilirsiniz. Aklınızda bulunsun Talya en çok şöminenin önünde uyumayı sever, sıcağa bayılır!!

Etkiler: Uyku 2 birim artar, huzur 2 birim azalır.

Duruma Göre Ek Etkiler: Sahibinin yanında uyuma seçeneği seçilirse ek bir mesaj verilir ve uyku 2 birim artar, kulübede uyuma seçeneği seçilirse uyku 2 birim artar, şöminenin yanındaki yatağa geçme seçeneği seçilirse uyku 2 birim artar.

2.4.3-geziVakti()

Bu fonksiyon, köpeğin gezme eylemini simüle ediyor. Bu gezme eylemi sosyalleşmesinden yanı sıra tuvalet ihtiyacını karşılamasına olanak tanıyor. Köpeğin hijyen düzeyi 3, uyku düzeyi 1 azalıyor; eğlence sosyallik ve huzur 1 artıyor.

2.5-Sağlık ihtiyacı

Köpeğimiz Talya'nın sağlık değerleri oldukça iyi, bir hastalığı yok fakat yine de dikkat etmezseniz hastalanabilir ve yaşamını yitirebilir. Talya'nın sağlıklı bir yaşam sürmesi için bazı fonksiyonlarımız bulunmaktadır:

2.5.1-veterinereGitmek()

Bu fonksiyon Talya'nın hasta olması durumunda sahibinin onu veterinere götürmesini simüle ediyor. Veterinere gittiğinde ilaç alarak ya da iğne yaptırarak iyileşen Talya'nın sağlık ve huzur seviyesinde artış, mikroplu bir ortam olduğundan dolayı hijyen seviyesinde azalış gözlemleniyor.

2.5.2-cezaVermek()

Bu fonksiyon Talya hatalı davranış yaptığında seçilir ve Talya'ya ceza verilir. Mama yemesi belirli bir süre yasaklanır veya eve girmesine izin verilmez. Seçilen ceza sonucunda Talya ya aç kalır ya da eve girmesi yasak olduğu için hastalanır. Her iki durumda da tokluk, sağlık ve huzur seviyelerinde azalış meydana gelir.

2.5.3-genclikIksiri()

Bu fonksiyon Talya'nın zaman zaman kullandığı gençlik iksirini içmesini simüle eden fonksiyondur. Bu işlemin kullanıcı tarafından seçilmesi durumunda iksirler arası seçim yapılması istenir: Sağlık iksiri, bilgecan dede iksiri, tokluk iksiri.

Sağlık iksiri, Talya'nın sağlık, hijyen ve huzur değerlerini 10'a çıkartıyor.

Bilgecan dede iksiri Talya'nın en sevdiği iksir. Sağlık, hijyen, huzur, tokluk, sevgi, uyku düzeyleri fulleniyor, tuvalet ihtiyacı gideriliyor.

Tokluk iksiri Talya'nın tokluk ve huzur düzeylerini 10'a çıkartıyor.

2.6-Eğlence ihtiyacı

Talya dünya üzerindeki en enerjik köpek olabilir! Oyun oynamak için her zaman enerjisi vardır. Eğlence ihtiyacını karşılamak için oluşturduğumuz 3 fonksiyonumuz da Talya'nın en sevdiği aktiviteleri belirtiyor.

2.6.1-sahibiIleVakitGecirme()

Bu fonksiyon Talya'nın sahibiyle vakit geçirmesini simüle ediyor. Talya sahibiyle oyun oynarken çok mutlu oluyor. Eğlence, sevgi, huzur seviyelerinde artış meydana geliyor. Yorulduğundan dolayı uyku seviyesinde artış, tokluk seviyesinde azalış meydana geliyor.

2.6.2-digerHayvanlarlaVakitGecirme()

Bu fonksiyon Talya'nın diğer hayvanlarla vakit geçirmesini simüle ediyor. Birlikte kemik yarışı yaptığı arkadaşlarıyla güzel vakit geçiriyor. Sevgi, eğlence ve huzur değerlerinde artış meydana geliyor. Sağlık değeri her fonksiyonun sonucunda olduğu gibi azalıyor.

2.6.3-denizdeYuzmek()

Bu fonksiyon Talya'nın sahibiyle birlikte denize gitmesini ve yüzmesini simüle ediyor. Talya suyu çok seven bir köpek, saatlerce suda oyun oynayabilir bu sebeple çok enerji harcar. Bu fonksiyon sonucunda Talya sevgi, eğlence, huzur ve uyku düzeylerinde artış, hijyen ve tokluk düzeylerinde ise azalış meydana geliyor.

2.7-Hijyen ihtiyacı

Önceki ihtiyaç fonksiyonlarının açıklamasında da belirttiğimiz gibi Talya oyun oynamayı, gezmeyi, eğlenmeyi çok seven bir köpek, Bu aktiviteler sonucunda da Talya sık sık kirleniyor ve temizlenmesi gerekiyor. Bu ihtiyaca ait 3 fonksiyon Talya'nın temiz bir köpek olmasını sağlamak için yazıldı.

2.7.1-dusAlmak()

Bu fonksiyon Talya'nın duş almasını simüle ediyor. Duş almasının sonucunda hijyen,sağlık, huzur ve uyku seviyelerinde artış meydana geliyor.

2.7.2-bakim()

Bu fonksiyon Talya'nın kişisel bakım ihtiyaçlarını simüle ediyor. Tırnak kesimi, tıraş olması ya da genel bir bakımdan söz edilebilir. Sağlık, hijyen ve huzur değerleri artış gösterir.

2.7.3-camurdaOynamak()

Bu fonksiyon Talya'nın çamurda oynamasını simüle ediyor. Hijyen ve sağlık değerleri azalırken huzur değerinde artış gözlenir.

2.8-Tuvalet ihtiyacı

Talya'nın tuvalet ihtiyacını gidermesi için 3 fonksiyon içerir.

2.8.1-ihtiyacYuruyusu()

Talya'nın ihtiyacını gidermesi için dışarı çıkartılıp yürüyüş yaptırılması gerektiğini anlatan fonksiyondur. Kullanıcı bu aktiviteyi seçtikten sonra fonksiyon 2 farklı case'e ayrılır.

Kullanıcıdan Talya'nın tuvalet ihtiyacını gidermesi için yer seçmesi istenir. Yer ağacın dibi ya da kum olabilir. Tuvalet ihtiyacı ve hijyen seviyesi azalır, huzur seviyesinde artış gözlenir.

2.8.2-tuvaleteCikma()

Bu fonksiyon Talya'nın evin içerisinde tuvalete çıkmasını simüle eder. Tuvalet ihtiyacı ve hijyen seviyesi azalır, huzur seviyesinde artış gözlenir.

2.8.3-tuvaletEgitimi()

Bu fonksiyon Talya'nın tuvalet eğitimi almasına yardımcı olur. Tuvalet ihtiyacını azaltır, eğitim düzeyini artırır.

2.9-Eğitim ihtiyacı

Talya'nın eğitime ihtiyacı olan konularda kullanıcı bu 3 fonksiyondan birini seçerek Talya'nın o konudaki ihtiyacını gidermesini sağlayabilir.

2.9.1-hocaylaEgitimeGitme()

Bu fonksiyon Talya'nın eğitmeniyle belirli konularda eğitim almasına olanak sağlar. Tokluk, eğlence ve uyku seviyesinde azalış gözlenir. Eğitim ve sosyallik seviyelerinde ise artış meydana gelir.

2.9.2-rehberlikEgitimi()

Bu fonksiyon Talya'nın engeli olan insanlara yardımcı olması amacıyla rehber köpek olması için verilen eğitimi simüle ediyor. Tokluk, eğlence ve uyku seviyesinde azalış gözlenir. Eğitim ve sosyallik seviyelerinde ise artış meydana gelir.

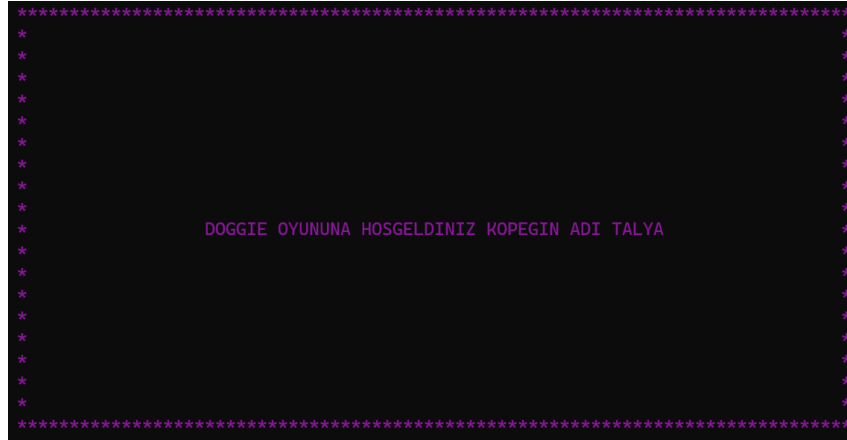
2.9.3-korumaEgitimi()

Bu fonksiyon Talya'nın sahibini dışarıdan gelebilecek her türlü tehlikeye karşı korumasını simüle ediyor. Tokluk, eğlence ve uyku seviyesinde azalış gözlenir. Eğitim ve sosyallik seviyelerinde ise artış meydana gelir.

2.10-Huzur düzeyi

Talya'nın belli başlı durumlarda huzurunun artıp veyahur huzursuzluğunun artacağını simüle eder.

3-ÇALIŞMA DETAYI



Şekil 3.1 : Karşılama ekranı görseli.

Kodumuz çalıştırıldığı an konsola for döngüsüyle kurduğumuz yıldızla yaratılan şekil ve içinde karşılama metnimiz beliriyor.

```
void drawBox(int width, int height, const char* text) {  
    for (int i = 0; i < height; i++) {  
        for (int j = 0; j < width; j++) {  
            if (i == 0 || i == height - 1 || j == 0 || j == width - 1) {  
                printf("*");  
            } else if (i == height / 2 && j == (width - strlen(text)) / 2) {  
                printf("%s", text);  
                j += strlen(text) - 1;  
            } else {  
                printf(" ");  
            }  
        }  
        printf("\n");  
    }  
}
```

Şekil 3.2 : Karşılama metni kodu

Karşılama metni kodunun çalışma mantığı sırasıyla aşağıdaki şekildedir:

1-void drawBox(int width, int height, const char* text): Fonksiyon, üç parametre alır: width (genişlik), height (yükseklik), ve text (kutu içine yerleştirilecek metin).

2-for (int i = 0; i < height; i++): İlk döngü, kutunun yüksekliği boyunca çalışır.

3-for (int j = 0; j < width; j++): İkinci iç içe döngü, kutunun genişliği boyunca çalışır.

4-if (i == 0 || i == height - 1 || j == 0 || j == width - 1): Bu koşul, kenarları çizmek için kullanılır. Eğer i (sıra) sıfır veya yüksekliğin bir eksiği veya j (sütun) sıfır veya genişliğin bir eksiği ise, bu durumda bir yıldız karakteri (*) yazdırılır.

5-else if (i == height / 2 && j == (width - strlen(text)) / 2): Bu koşul, kutu içine metni eklemek içindir. Eğer i orta satıra (yüksekliğin yarısı) ve j metnin kutu içindeki konumunun başlangıcına denk geliyorsa, printf("%s", text) kullanılarak metin kutu içine yazdırılır. Ardından, j değeri, yazdırılan metnin uzunluğu kadar artırılır.

6-else: Yukarıdaki koşullar sağlanmazsa, bu durumda bir boşluk karakteri yazdırılır.

7-printf("\n");: Her satırın sonunda bir alt satıra geçmek için kullanılır.

8-Fonksiyon, genişliği ve yüksekliği belirtilen bir kutu çizer. Eğer text parametresi verilmişse, bu metni kutunun ortasına yerleştirir. Ancak, metni yerleştirirken j değeri artırılarak bu işlem gerçekleştirildiğinden, metni yerleştirdikten sonra dış döngünün devamında j değeri daha fazla artırılmaz. Bu, metnin kutu dışına taşmasını önler.

```
DUZEYLER
tokluk duzeyi= 5
sevgi duzeyi= 5
tuvalet duzeyi= 5
hijyen duzeyi= 5
egitim duzeyi= 5
uyku duzeyi= 5
eglence duzeyi= 5
sosyallik duzeyi= 5
saglik duzeyi= 5
huzur duzeyi= 5
```

Şekil 3.3 : Düzeyler görseli

Bütün ihtiyaçların düzeyleri while içinde for döngüsüyle sırasıyla ekrana yazdırılıyor.

```
while(1){
    printf("\n\n");

    printf("DUZEYLER\n");
    for(int i=0;i<10;i++){
        printf("%s duzeyi= %d\n",isimler[i],duzeyler[i]);
    }
}
```

Şekil 3.4 : Düzeyler kısmını yazdıran kodun görseli

Bu program, bir sonsuz döngü içinde bulunuyor (while(1)), bu da demek oluyor ki, program sürekli olarak çalışmaya devam edecek. Menü siz çıkış yapana kadar ekrana yazdırılmaya devam edecek.

Bir döngü içinde, isimler dizisi ve duzeyler dizisi kullanılarak 10 elemanlı bir liste yazdırılıyor. Her bir isimler elemanı ile karşılık gelen duzeyler elemanı eşleştirilip ekrana yazdırılıyor (printf("%s duzeyi= %d\n", isimler[i], duzeyler[i]));).

```
1. islem = Sahibi Tarafından Sevmek
2. islem = Yabancı Biri Tarafından Sevmek
3. islem = Karsi Cins Tarafından Sevmek
4. islem = Uyku Vakti
5. islem = Zorla Uyandır
6. islem = Gezi Vakti
7. islem = Yemek yemek
8. islem = Su içmek
9. islem = Yeni Lezzetler Denemek
10. islem = Kopek Parkında Vakit Gecirme
11. islem = Evde Durma
12. islem = Asik Olmak
13. islem = Veterinere Gitmek
14. islem = Ceza Vermek
15. islem = Genclik Iksiri
16. islem = Sahibi Ile Vakit Gecirme
17. islem = Diger Hayvanlarla Vakit Gecirme
18. islem = Denizde Yuzme
19. islem = Dus Almak
20. islem = Kisisel Bakim
21. islem = Camurda Oynamak
22. islem = Ihtiyac Yurusu
23. islem = Tuvalete Cikma
24. islem = Tuvalet Egitimi
25. islem = Hocayla Egitime Gitme
26. islem = Rehberlik Egitimi
27. islem = Koruma Egitimi
28. islem = cikis
```

Şekil 3.5 : Menü görseli

```
for (int i = 0; i <=27; i++) {
    printf("%d. islem = %s\n", i+1, ihtiyac[i] );
}
```

Şekil 3.6 : Menüü ekrana yazdırma kodu

Döngü içinde, ihtiyac dizisi kullanılarak 28 elemanlı bir liste yazdırılıyor (printf("%d. islem = %s\n", i + 1, ihtiyac[i]));).

```
Lutfen giderilecek ihtiyaci girin.1-28 arasinda bir deger giriniz. (cikis icin 28 e basiniz.):
```

Şekil 3.7 : Kullanıcıdan seçenek isteme görseli

```
int indis;

printf("Lutfen giderilecek ihtiyaci girin.1-28 arasinda bir deger giriniz. (cikis icin 28 e basiniz.): ");
scanf("%d", &indis);

if (indis == 28) {
    printf("Kendine iyi bak.Seni ozleyecegim...\n");
    return 0;
} else if (indis < 1 || indis > 28) {
    printf("Gecersiz bir secim yaptiniz. Lutfen tekrar deneyin.\n");
    continue;
}
```

Şekil 3.8 : Kullanıcıdan seçenek isteme kodu

Bu kısım, kullanıcıdan bir girdi almak ve bu girdiye göre işlemler yapmak için kullanılmaktadır. Kullanıcıdan bir girdi istenmektedir. Kullanıcıya 1 ile 28 arasında bir sayı girmesi gerektiği ve programdan çıkmak için 28'i girmesi gerektiği hatırlatılmaktadır. Eğer kullanıcı 28'i girerse, program "Kendine iyi bak.Seni ozleyecegim..." yazısını ekrana yazdırır

ve programı sonlandırır (return 0;). Eğer kullanıcı 1 ile 28 arasında bir sayı giremezse, yani geçersiz bir giriş yaparsa, program kullanıcıya "Gecersiz bir secim yaptiniz. Lutfen tekrar deneyin." yazısını ekrana yazdırır ve bir sonraki döngü adımına geçer (continue;).

```
int sevgi;
int sosyallik;
int eglenme;
int uyku;
int saglik;
int hijyen;
int tuvalet;
int egitim;
int tokluk;
int huzur;

char *isimler[10] = {"tokluk", "sevgi",
"tuvalet", "hijyen", "egitim", "uyku",
"eglenme", "sosyallik", "saglik", "huzur"};
int duzeyler[10] = {5, 5, 5, 5, 5, 5, 5, 5, 5, 5};
```

Şekil 3.9 : Tanımlanan verilerin kod görseli

Kodumuzda esas aldığımız 10 ihtiyaç integer değer olarak kodumuzun en başında tanımlanıyor. İsimler: 10 elemanlı bir karakter dizisi. Her biri bir ihtiyacı temsil eder, duzeyler: Her bir ihtiyacın seviyesini tutan bir dizi. Başlangıçta her ihtiyaç için 5 olarak başlıyor. Sonrasında menüden seçilen eylemlere göre düzeyler artış ya da azalış gösteriyor.

```
void geziVakti(int *psayilar){
    printf("Sanirim yine gezmeye cikiyoruz.Simdiden esnemeye basladim...\n");
    psayilar[3]=psayilar[3]-2;
    psayilar[6]=psayilar[6]+2;
    psayilar[7]=psayilar[7]+2;
    psayilar[5]=psayilar[5]-1;
    psayilar[9]=psayilar[9]+2;

    for(int i=0;i<10;i++){
        if(i==8 || i==5){
            psayilar[i]= psayilar[i];
        }
        else{
            psayilar[i]= psayilar[i]-1;
        }
    }
    kontrolEt(duzeyler);
}
```

Şekil 3.10 : Basit bir fonksiyon kod örneği

Bu kod örneği gibi 28 tane fonksiyon kodumuz bulunmakta. Kullanıcıdan aldığımız her komut için seçilen komutun fonksiyonu çalışıyor. Görsel 3.10'daki fonksiyon Talya'nın gezi vakti fonksiyonu. Eğer kullanıcı 28 seçenek bulunan menüde 6 numarayı seçerse bu fonksiyon çalışıyor. Fonksiyon başlangıçta mainin içindeki switch case yapısındaki ilgili case'e yönlendiriliyor. Case'in içinde çağırılan fonksiyonun içerisine düzeyler dizisini yazıyoruz. Düzeyler dizisinin elemanlarını pointer yardımıyla psayilar dizisine eşitliyoruz. Ve elimizdeki tüm fonksiyonlar parametre olarak psayilar dizisini alıyor. *psayilar parametresiyle her işlem yapıldıktan sonra düzeyler tekrardan yazılıyor.

```

void kontrolEt(int *duzeyler) {
    for(int i = 0; i < 10; i++) {
        if (duzeyler[i] > 10) {
            duzeyler[i] = 10;
        } else if (duzeyler[i] < 0) {
            duzeyler[i] = 0;
        }
    }
}

```

Şekil 3.11 : Kontrol fonksiyonu kodu

Her fonksiyonumuzun içinde bulunan kontrolEt() fonksiyonu *duzeyler dizisini parametre olarak alır. Pointer yardımıyla adreslerine ulaşır, adreslerin içerisinde bulunan değerlerin 0 ile 10 arasında olup olmadığını kontrol eder. Eğer ilgili düzey 10'dan büyükse 10'a eşitlenir, 0'dan küçükse 0'a eşitlenir. Bu sayede düzeyleri 0 ile 10 arasında tutabiliyoruz.

```

void YeniLezzetlerDenemek(int *psayilar){
    int yeni;

    printf("Yeni ve de farklı tatlara acıgiz galiba!Ama sen yine de dikkat et,biraz hassassin.\n");
    printf("Ne denemek istersin?\n1-Ananas\n2-Uzak dogu yoresel kopek yemegi\n");
    scanf("%d",&yeni);

    switch(yeni){
    case 1:
        printf("Bu cok guzel bir sey mis.Neden daha once denemedik?\n\n");

        psayilar[0]= psayilar[0]+3;
        psayilar[8]= psayilar[8]+1;

        break;

    case 2:
        printf("Aman Tanrim,FELAKET.Sanirim kusacagim.");

        psayilar[0]= psayilar[0]-1;
        psayilar[8]= psayilar[8]-2;

        break;
    }

    for(int i=0;i<10;i++){
        if(i==8 || i==0){
            psayilar[i]= psayilar[i];
        }
        else{
            psayilar[i]= psayilar[i]-1;
        }
    }

    kontrolEt(duzeyler);
}

```

Şekil 3.12 : Switch case yapısı bulunduran fonksiyon kodu örneği

Kullanıcı menüden 9 numaralı seçeneği seçtiğinde yeniLezzetlerDenemek adlı fonksiyon çağırılıyor. Bu fonksiyon kendi içerisinde 2 farklı case'e ayrılıyor. Kullanıcı Talya'nın hangi yiyeceği yemek istediğini caseler aracılığıyla seçiyor ve seçilen caselere göre ilgili metin ekrana yazdırılıyor, son olarak güncellenen düzeyler ekrana yazdırılıyor.

```

switch(indis-1) {

    case 0:
        printf("Giderilek ihtiyaç: %s\n", ihtiyaç[0]);
        sahibiTarafindanSevilmek(duzeyler);
        kontrolEt(duzeyler);
        break;

    case 1:
        printf("Giderilek ihtiyaç: %s\n", ihtiyaç[1]);
        yabancıBiriTarafindanSevilmek(duzeyler);
        kontrolEt(duzeyler);
        break;

    case 2:
        printf("Giderilek ihtiyaç: %s\n", ihtiyaç[2]);
        karsiCinsTarafindanSevilmek(duzeyler);
        kontrolEt(duzeyler);
        break;

    case 25:
        printf("Giderilek ihtiyaç: %s\n", ihtiyaç[25]);
        rehberlikEgitimi(duzeyler);
        kontrolEt(duzeyler);
        break;

    case 26:
        printf("Giderilek ihtiyaç: %s\n", ihtiyaç[26]);
        korumaEgitimi(duzeyler);
        kontrolEt(duzeyler);
        break;

    case 27:
        printf("cikis yapilmistir.gule gule...tekrar bekleriz\n", ihtiyaç[27]);
        kontrolEt(duzeyler);
        break;
}

```

Şekil 3.13 : Case örnekleri

Kullanıcının menüden seçtiği aktivitenin numarası mainin içindeki aynı indise sahip case'i çalıştırır. Case'in içerisindeki fonksiyona gidilir, fonksiyon çalıştırıldıktan sonra kontrolEt fonksiyonu çalıştırılır. Düzeyler bu fonksiyon içerisinde kontrol edildikten sonra tekrardan menü çağırılır. Çıkış yapılan kadar kullanıcıya menü tekrar sunulur, işlem yapması beklenir. Çıkış case'i açıldığında kullanıcı oyundan çıkarılır.

```

if (duzeyler[8]<=0){
    for(int i=0;i<9;i++){
        printf("%s duzeyi= %d\n", isimler[i],duzeyler[i]);
    }
    printf("Talya yasam fonksiyonlarini vitirdi.Oyun sonlandi.\n ");
    return 0;
}

if(duzeyler[0]>=10){
    for(int i=0;i<9;i++){
        printf("%s duzeyi= %d\n", isimler[i],duzeyler[i]);
    }
    printf("Talya fazla yemekten zehirlendi ve yasamini vitirdi.Oyun sonlandi.\n ");
    return 0;
}

if(duzeyler[0]==0 && duzeyler[5]<=0){
    for(int i=0;i<9;i++){
        printf("%s duzeyi= %d\n", isimler[i],duzeyler[i]);
    }
    printf("Talya aclik ve uykusuzluktan yasamini vitirdi.Oyun sonlandi.\n ");
    return 0;
}

if(duzeyler[0]==0){
    for(int i=0;i<9;i++){
        printf("%s duzeyi= %d\n", isimler[i],duzeyler[i]);
    }
    printf("Talya aclikten yasamini vitirdi.Oyun sonlandi.\n ");
    return 0;
}

if(duzeyler[5]<=0 ){
    for(int i=0;i<9;i++){
        printf("%s duzeyi= %d\n", isimler[i],duzeyler[i]);
    }
    printf("Talya uykusuzluktan yasamini vitirdi.Oyun sonlandi.\n ");
    return 0;
}

```

Şekil 3.14 :Oyunun sonlanma koşulları

Oyunumuzda kullanıcı tarafından seçilen bazı aktivitelerin sonucunda düzeyler artış ya da azalış gösterir. Bu artış ve azalışın belirli fonksiyonlarda belirlenen sınırlar dışında olması durumunda return yapısıyla oyunun sonlanması sağlanır. Örneğin Şekil 3.14'teki birinci if yapısında da görüldüğü üzere tokluk seviyesi 0 ın altına veya 10 un üzerine çıktığı durumlarda açlıktan veya zehirlenmeden dolayı yaşam fonksiyonlarını yitirir bunun sonucu olarak oyundan çıkış yapılır.

```
for(int i=0;i<10;i++){  
    if(i==8 || i==4){  
        psayilar[i]= psayilar[i];  
    }  
    else{  
        psayilar[i]= psayilar[i]-1;  
    }  
}
```

Şekil 3.15 : Fonksiyon değerlerinin azaltılması

Çağrılan her fonksiyon kendi bulunduğu ihtiyaç değerini ve sağlık değeri haricindekileri 1 azaltır.

4-TEKNİK EKSİKLİKLER

Teknik bir eksikliğimiz olduğunu düşünmüyoruz. Projede istenen her şeyi yerine getirdik, yaratıcılık kısmını da ekledik.

5-YARATICILIK AŞAMASI

```
void asikOlmak(int *psayilar){  
    int asik;  
    printf("Sanırım birileri asik olmus.Bana soylesmek istedigim bir sayilar var mi?\n\n");  
    printf("1-EVET\n2-HAYIR");  
    scanf("%d",&asik);  
    switch(asik){  
    case 1:  
        printf("Sanırım yakalandım.Buddy'le gecen parkta oynadigimiz gun ben biraz ondan hoslanmis olabilirim.\n");  
        psayilar[1]=psayilar[1]+3;  
        psayilar[9]=psayilar[9]+2;  
        break;  
    case 2:  
        printf("Malesef kimseve asik degilim.Deniyorum fakat olmuyor.Bu biraz canimi sikiyor olabilir.Ve canim cikolata istiyor.\n");  
        psayilar[1]=psayilar[1]-2;  
        psayilar[9]=psayilar[9]-2;  
        psayilar[0]=psayilar[0]-2;  
        break;  
    }  
}
```

Şekil 5.1 : Aşık olmak fonksiyonu kod görseli

Kodumuza aşık olmak fonksiyonunu ekledik ve sahibiyile bu durumu konuşmak isteyip istememesini sordurduk.

```

void genclikIksiri(int *psayilar){
    int iksir;
    printf("Sanirim biraz takviyeye ihtiyacim var.\n\n");
    printf("Hangi iksirden icmek istiyorsun?\n");
    printf("1-Saglik iksiri\n2-Bilgecan dede iksiri\n3-Tokluk iksiri\n");
    scanf("%d",&iksir);

    switch(iksir){
        case 1:
            printf("Saglik iksiri icildi.Saglik,hiyen ve buzur duzeyleri fullendi.\n");

            psayilar[3]=10;
            psayilar[8]=10;
            psayilar[9]=10;

            break;

        case 2:
            printf("Bilgecan Dede iksiri icildi.Saglik,hiyen,buzur,tokluk,savvi,uvku duzeyleri fullendi.Tuvalet duzeyi sifirlandi.\n");
            psayilar[0]=10;
            psayilar[1]=10;
            psayilar[2]=0;
            psayilar[3]=10;
            psayilar[4]=10;
            psayilar[5]=10;
            psayilar[9]=10;
            break;

        case 3:
            printf("Tokluk iksiri icildi.Tokluk ve buzur duzeyleri fullendi.\n");
            psayilar[0]=10;
            psayilar[9]=10;

            break;
    }
}

```

Şekil 5.2 : Gençlik iksiri fonksiyonu kod görseli

Talya opsiyonel olarak artırmak istediği değerlere yönelik koddaki iksirlerden birini içip değerlerini iyileştirebilir.

```

void kopekParkinaGitmek(int *psayilar){
    int secim;
    int secim2;
    int secim3;
    //bas:
    printf("1-COCO\n2-BUDDY\n3-BELLA\n");
    printf("Hangi arkadasinla oynamak istersin?\n");
    scanf("%d",&secim);

    switch(secim){
        case 1:
            printf("Talya en yakin arkadası COCO'yla vakit geciriyor.\n");
            psayilar[1]=psayilar[1]+3;
            psayilar[7]=psayilar[7]+4;
            psayilar[6]=psayilar[6]+3;
            psayilar[5]=psayilar[5]-1;
            psayilar[9]=psayilar[9]+2;
            break;

        case 2:
            printf("Talya hoslandigi köpek ile vakit geciriyor.\nÖncesinde çok suslendi.\n");
            psayilar[1]=psayilar[1]+5;
            psayilar[7]=psayilar[7]+4;
            psayilar[6]=psayilar[6]+3;
            psayilar[5]=psayilar[5]-1;
            psayilar[9]=psayilar[9]+2;
            break;

        case 3:
            printf("Talya Bella'yi saymıyor.Talya'nın onunla vakit gecirmesini istedigine emin misin?\n");
            printf("1-EVET\n2-HAYIR\n");
            scanf("%d",&secim2);

            if (secim2==1){
                printf("Talya Bella'yla vakit gecirmeye istekli.\n");
                psayilar[1]=psayilar[1]+2;
                psayilar[7]=psayilar[7]+4;
                psayilar[6]=psayilar[6]+2;
                psayilar[5]=psayilar[5]-1;
            }
            else if (secim2==2){
                printf("Talya Bella'yla vakit gecirmek istemedigini hatirladi.\n");
                printf("1-COCO\n2-BUDDY\n");
                printf("Hangi arkadasinla oynamak istersin?\n");
                scanf("%d",&secim3);

                if (secim3==1){
                    printf("Talya en yakin arkadası COCO'yla vakit geciriyor.\n");
                    psayilar[1]=psayilar[1]+3;
                    psayilar[7]=psayilar[7]+4;
                    psayilar[6]=psayilar[6]+3;
                    psayilar[5]=psayilar[5]-1;
                    psayilar[9]=psayilar[9]+2;
                }

                else if (secim3==2) {
                    printf("Talya hoslandigi köpek ile vakit geciriyor.\nÖncesinde çok suslendi.\n");
                    psayilar[1]=psayilar[1]+5;
                    psayilar[7]=psayilar[7]+4;
                    psayilar[6]=psayilar[6]+3;
                    psayilar[5]=psayilar[5]-1;
                    psayilar[9]=psayilar[9]+2;
                }
            }
    }
}

```

Şekil 5.3 : Köpek parkına gitme fonksiyonu kod görseli

Talya köpek parkına gittiğinde sistem, kullanıcıya arkadaş seçimini yaptırır. Bu seçimde başlangıçta 3 arkadaş kullanıcıya sunulur. Kullanıcı arkadaşların arasından Bella'yı seçtiğinde sistem en içteki if else yapısını kullanarak 'emin misin?' sorusunu sorar. Kullanıcının gireceği evet ya da hayır değerine göre aktivite fonksiyonu sonlanır ve ekrana tekrar seçim yapmak için menü getirilir.

KAYNAKÇA

<https://stackoverflow.com>

<https://github.com>

<https://www.w3schools.com>

<https://chat.openai.com>