WHITEBOX TESTIRANJE – SetProgressData metoda za RegisteredUserController

1. METODA

```csharp
// Async method to set progress data for a registered user
3 references | we ran into an exception loading metrics for this method - please contact support | ● 1/1 passing
public async Task<List<List<object>>> SetProgressData(RegisteredUser registeredUser)
{
    // Current date information
    var currentDate = DateTime.Now.Date;
    var currentDayOfWeek = (int)currentDate.DayOfWeek;

    // Retrieve progress data for the last 7 days
    var progressList = await _context.Progress
        .Where(p => p.RegisteredUser.Id == registeredUser.Id && p.Date.Date >= currentDate.AddDays(-6).Date && p.Date.Date <= currentDate.Date)
        .OrderBy(p => p.Date)
        .ToListAsync();

    // Average consumed and burned calories values
    var averageConsumedCalories = 2000;
    var averageBurnedCalories = 300;

    // Lists to store consumed and burned calories progress data
    var consumedCaloriesProgressData = new List<object>();
    var burnedCaloriesProgressData = new List<object>();

    // Loop through the last 7 days to calculate progress data
    for (var i = 6; i >= 0; i--)
    {
        var date = currentDate.AddDays(-i);
        var dayOfWeek = (int)date.DayOfWeek;
        var progress = progressList.FirstOrDefault(p => p.Date.Date == date.Date);

        // Consumed and burned calories values for the day
        var consumedCalories = progress?.ConsumedCalories ?? 0;
        var burnedCalories = progress?.BurnedCalories ?? 0;

        // Calculate progress percentages
        var consumedCaloriesProgressPercentage = (int)CalculateProgressPercentage(consumedCalories, averageConsumedCalories);
        var burnedCaloriesProgressPercentage = (int)CalculateProgressPercentage(burnedCalories, averageBurnedCalories);
        var isSelectedDay = dayOfWeek == currentDayOfWeek;

        // Add progress data for the day to the respective lists
        consumedCaloriesProgressData.Add(new
        {
            DayOfWeek = date.ToString("ddd"),
            HeightPercentage = consumedCaloriesProgressPercentage,
            IsSelectedDay = isSelectedDay
        });

        burnedCaloriesProgressData.Add(new
        {
            DayOfWeek = date.ToString("ddd"),
            HeightPercentage = burnedCaloriesProgressPercentage,
            IsSelectedDay = isSelectedDay
        });
    }

    // Create a list to store both consumed and burned calories progress data
    var list = new List<List<Object>>();
    list.Add(consumedCaloriesProgressData);
    list.Add(burnedCaloriesProgressData);
    return list;
}
```

2. DIJAGRAM TOKA

```
                              ●

          currentDate = DateTime.Now.Date
      currentDateOfWeek = currentDate.DayOfWeek
                    progressList
               averageConsumedCalories
                averageBurnedCalories
                  consumedCalories
                    burnedCalories


            date = currentDate.AddDays(-i);
          dayOfWeek = (int)date.DayOfWeek;
                      progress


    consumedCalories==null
  consumedCalories=0  ◄──────◇
                             consumedCalories!=null

    burnedCalories==null
  burnedCalories=0  ◄──────◇
                             burnedCalories!=null


                       ◇  consumedCaloriesProgressPercentage

  progress>100      progress<0        progress>0 && progress<100
  progress=100                         progress=
               progress=0            value/averageValue*100

                       ◇


                       ◇  burnedCaloriesProgressPercentage

  progress>100      progress<0        progress>0 && progress<100
  progress=100                         progress=
               progress=0            value/averageValue*100

                       ◇


          isSelectedDay = dayOfWeek ==
                  currentDayOfWeek
      consumedCaloriesProgressData.Add(progress)
      burnedCaloriesProgressData.Add(progress)


          list = new List<List<Object>>();
      list.Add(consumedCaloriesProgressData);
       list.Add(burnedCaloriesProgressData);
```

3. GRAF PROGRAMSKOG TOKA
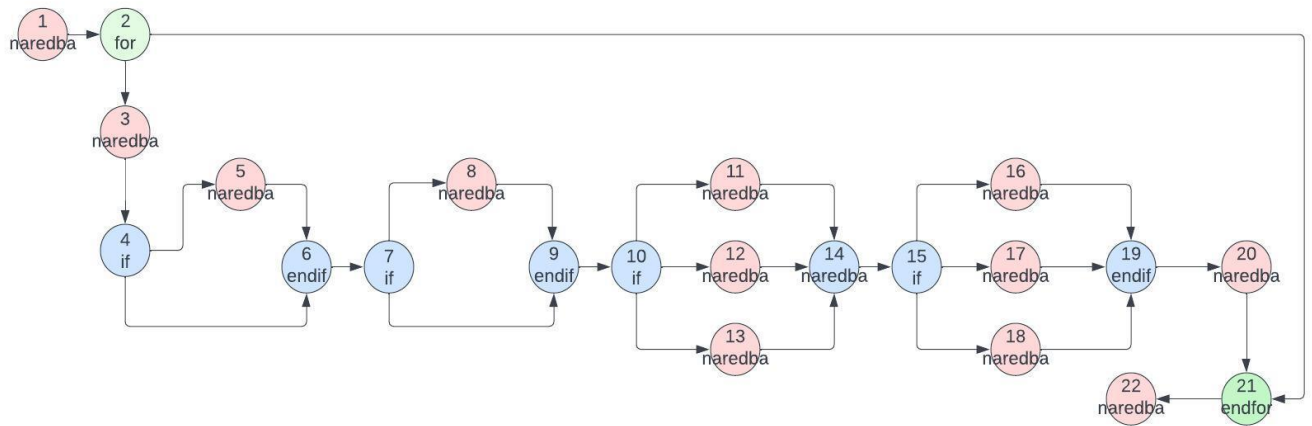
broj grana - broj čvorova + 2 = 28 - 22 +2 = 8

4. PUTEVI:

1-2-3-4-5-6-7-9-10-11-14-15-18-19-20-21-22

1-2-3-4-6-7-8-9-10-13-14-15-17-19-20-21-22

1-2-3-4-4-6-7-9-10-12-14-15-17-19-20-21-22

1-2-3-4-5-6-7-8-9-10-11-14-15-16-19-20-21-22

1-2-3-4-5-6-7-8-9-10-11-14-15-17-19-20-21

1-2-3-4-5-6-7-8-9-10-11-14-15-18-19-20-21-22

1-2-3-4-5-6-7-8-9-10-13-14-15-16-19-20-21-22

1-2-3-4-5-6-7-8-9-10-1-12-14-15-17-19-20-21-22

1-2-3-4-5-6-7-8-9-10-12-14-15-18-19-20-21-22

1-2-3-4-5-6-7-8-9-10-13-14-15-16-19-20-21-22

1-2-3-4-5-6-7-8-9-10-13-14-15-17-19-20-21-22

1-2-3-4-5-6-7-8-9-10-13-14-15-18-19-20-21-22

5. TESTNA METODA

```csharp
[TestMethod]
[DynamicData("ProgressData")]
● | 0 references | we ran into an exception loading metrics for this method - please contact support
public async Task Index_ReturnsViewResultForDifferentValuesOfProgress_WithCorrectModelAndProgressData(List<NutritionTipsAndQuotes> nutritionTips,
    RegisteredUser registeredUser, Progress progressData)
{

    _mockDbContext.Setup(c => c.Progress).ReturnsDbSet(new List<Progress>());
    _mockDbContext.Setup(db => db.NutritionTipsAndQuotes).ReturnsDbSet(nutritionTips);

    // Act
    var result = await _controller.SetProgressData(registeredUser);

    // Assert
    Assert.IsNotNull(result);
    // Check if lists of consumed and burnt calories are both filled with data
    Assert.AreEqual(2, result.Count);

}
```

S obzirom da se testni primjeri razlikuju samo u ulaznim podacima vezanim za Progress, odabran je data driven način testiranja, te su testni slučajevi definirani na sljedeći način:

```csharp
0 references
public static IEnumerable<object[]> ProgressData
{
    get
    {
        var nutritionTips = new List<NutritionTipsAndQuotes>
        {
            new NutritionTipsAndQuotes
            {
                NTAQId= 1,
                QuoteText="Quote1",
            },
            new NutritionTipsAndQuotes
            {
                NTAQId = 2,
                QuoteText="Quote2",
            }
        };

        var registeredUser = new RegisteredUser { Id = "userId" };

        return new[]
        {
            new object[] { nutritionTips, registeredUser, new Progress { Date = DateTime.Now.Date.AddDays(-6), ConsumedCalories = 1500, RegisteredUser = registeredUser } },
            new object[] { nutritionTips, registeredUser, new Progress { Date = DateTime.Now.Date.AddDays(-5), BurnedCalories = -200, RegisteredUser = registeredUser } },
            new object[] { nutritionTips, registeredUser, new Progress { Date = DateTime.Now.Date.AddDays(-5), RegisteredUser = registeredUser } },
            new object[] { nutritionTips, registeredUser, new Progress { Date = DateTime.Now.Date.AddDays(-5), ConsumedCalories = 1500, BurnedCalories = 400, RegisteredUser = registeredUser } },
            new object[] { nutritionTips, registeredUser, new Progress { Date = DateTime.Now.Date.AddDays(-5), ConsumedCalories = 1500, BurnedCalories = -200, RegisteredUser = registeredUser } },
            new object[] { nutritionTips, registeredUser, new Progress { Date = DateTime.Now.Date.AddDays(-4), ConsumedCalories = 1500, BurnedCalories = 50, RegisteredUser = registeredUser } },
            new object[] { nutritionTips, registeredUser, new Progress { Date = DateTime.Now.Date.AddDays(-3), ConsumedCalories = -500, BurnedCalories = 200, RegisteredUser = registeredUser } },
            new object[] { nutritionTips, registeredUser, new Progress { Date = DateTime.Now.Date.AddDays(-2), ConsumedCalories = -500, BurnedCalories = -50, RegisteredUser = registeredUser } },
            new object[] { nutritionTips, registeredUser, new Progress { Date = DateTime.Now.Date.AddDays(-2), ConsumedCalories = -500, BurnedCalories = 50, RegisteredUser = registeredUser } },
            new object[] { nutritionTips, registeredUser, new Progress { Date = DateTime.Now.Date.AddDays(-1), ConsumedCalories = 50, BurnedCalories = 200, RegisteredUser = registeredUser } },
            new object[] { nutritionTips, registeredUser, new Progress { Date = DateTime.Now.Date.AddDays(0), ConsumedCalories = 50, BurnedCalories = -50, RegisteredUser = registeredUser } },
            new object[] { nutritionTips, registeredUser, new Progress { Date = DateTime.Now.Date.AddDays(0), ConsumedCalories = 50, BurnedCalories = 50, RegisteredUser = registeredUser } },
        };
    }
}
```

Ovim testom su obuhvaćene sve linije, petlje i putevi metode.