PROJECT BIG DATA PROCESSING

ANALYSIS STUDENTS PERFORMANCE IN EXAM

Christian Joseph R. - 2440050311
Steven Nathaniel S - 2440058403
Raysen Harison B - 2401959105
Yoseph Kurniawan - 2440064854

Kelas: LC01 - LEC

LATAR BELAKANG

Dalam proses pembelajaran, terkadang performa belajar siswa naik turun. Menganalisis performa siswa adalah aspek penting dari proses pengajaran. Performa siswa dapat mempengaruhi hasil yang diraih oleh siswa. Performa siswa tersebut dipengaruhi oleh berbagai keadaan, termasuk latar belakang pendidikan orang tua, persiapan ujian, kesehatan siswa, dan sebagainya. Ujian berfungsi sebagai penilaian akhir siswa, memungkinkan mereka untuk mengetahui apa yang telah mereka pelajari selama studi mereka. Oleh karena itu, dengan dataset yang telah kami cari, kami akan menggunakan dataset ini untuk menganalisis faktor-faktor yang dapat mempengaruhi performa siswa terutama dalam menghadapi ujian dan menentukan faktor yang paling mempengaruhi performa siswa.

RUMUSAN MASALAH

1. Apa saja faktor yang mempengaruhi nilai siswa?
2. Apa faktor yang paling mempengaruhi performa siswa?

TUJUAN

1. Untuk mengetahui faktor yang dapat mempengaruhi nilai siswa.
2. Untuk mengetahui faktor yang paling mempengaruhi performa siswa

PENJELASAN DATASET

Kami menggunakan dataset ini karena dataset ini telah menyediakan data yang berupa faktor yang kami ingin analisis mengenai faktor-faktor yang dapat mempengaruhi performa siswa/i khususnya dalam persiapan menghadapi ujian.

Dataset Performance Student in Exam:

https://www.kaggle.com/datasets/spscientist/students-performance-in-exams

Google Collab:

https://colab.research.google.com/drive/19lqIcTqc8QtOsQjr0nvTAuNM1SkCG3eT?usp=sharing

TIPE DATA

```
Data columns (total 11 columns):
 #   Column                       Non-Null Count  Dtype  
---  ------                       --------------  -----  
 0   gender                       1000 non-null   object 
 1   race/ethnicity               1000 non-null   object 
 2   parental level of education  1000 non-null   object 
 3   lunch                        1000 non-null   object 
 4   test preparation course      1000 non-null   object 
 5   math score                   1000 non-null   int64  
 6   reading score                1000 non-null   int64  
 7   writing score                1000 non-null   int64  
 8   Total score                  1000 non-null   int64  
 9   Percentage                   1000 non-null   float64
 10  grade                        1000 non-null   object 
dtypes: float64(1), int64(4), object(6)
```

SAMPLE DATA

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score	Total score	Percentage	grade
0	female	group B	bachelor's degree	standard	none	72	72	74	218	72.666667	B-
1	female	group C	some college	standard	completed	69	90	88	247	82.333333	A-
2	female	group B	master's degree	standard	none	90	95	93	278	92.666667	A
3	male	group A	associate's degree	free/reduced	none	47	57	44	148	49.333333	F
4	male	group C	some college	standard	none	76	78	75	229	76.333333	B
...	...	...	...	...	...	...	...	...	...	...	...
995	female	group E	master's degree	standard	completed	88	99	95	282	94.000000	A
996	male	group C	high school	free/reduced	none	62	55	55	172	57.333333	D
997	female	group C	high school	free/reduced	completed	59	71	65	195	65.000000	C
998	female	group D	some college	standard	completed	68	78	77	223	74.333333	B-
999	female	group D	some college	free/reduced	none	77	86	86	249	83.000000	A-

Attribute Information

gender: Male/ Female

race/ethnicity: Group division from A to E

parental level of education: Details of parental education varying from high school to master's degree

lunch: Type of lunch selected

test preparation course: Course details

math score: Marks secured by a student in Mathematics

reading score: Marks secured by a student in Reading

writing score: Marks secured by a student in Writing

PROSES DATA

GRADING SYTEMS

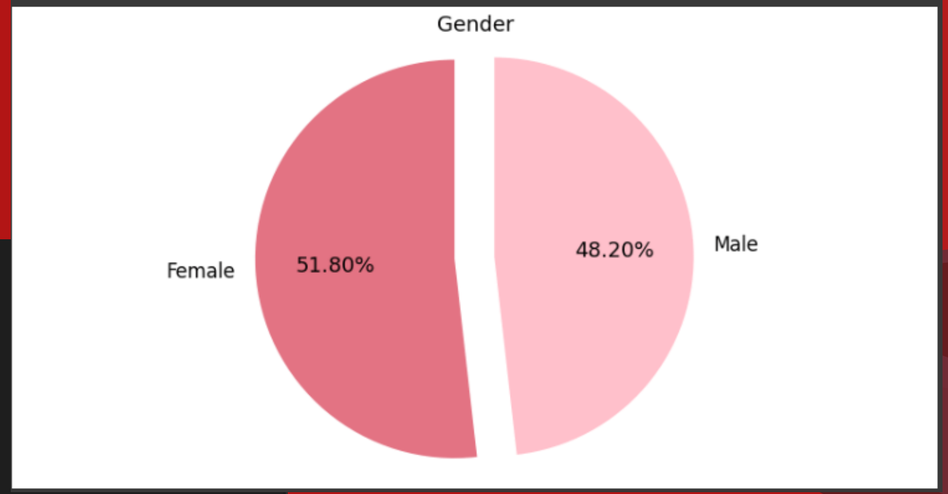
```
def Grade(Percentage):
    if (Percentage >= 85):return 'A'
    if (Percentage >= 80):return 'A-'
    if (Percentage >= 75):return 'B'
    if (Percentage >= 70):return 'B-'
    if (Percentage >= 65):return 'C'
    if (Percentage >= 60):return 'C-'
    if (Percentage >= 55):return 'D'
    if (Percentage >= 50):return 'E'
    else: return 'F'

data["grade"] = data.apply(lambda x : Grade(x["Percentage"]), axis=1)
```

FACTOR GENDER

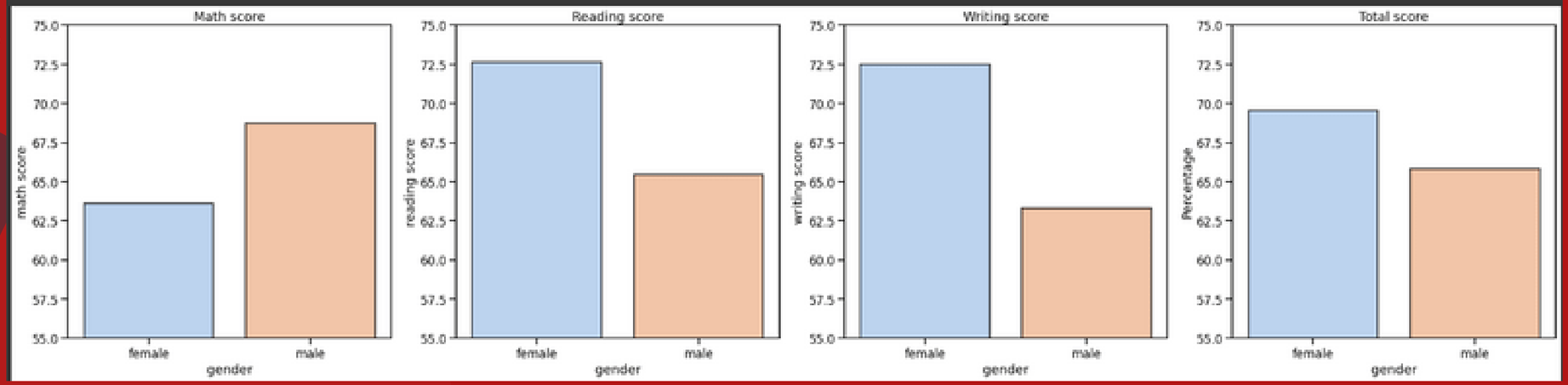
```
#show count Gender
data['gender'].value_counts()
```

```
female    518
male      482
Name: gender, dtype: int64
```

```
plt.figure(figsize=(14, 7))
labels=['Female', 'Male']
plt.pie(data['gender'].value_counts(), labels=labels, explode=[0.1,0.1],
        autopct='%1.2f%%',colors=['#E37383','#FFC0CB'], startangle=90)
plt.title('Gender')
plt.axis('equal')
plt.show()
```




```
fig , axes = plt.subplots(1,4, figsize=(37,8))
sns.barplot(data = data, x='gender', y='math score',edgecolor='#0000',**{'alpha':0.8,'linewidth':2},ax = axes[0],ci=None)
axes[0].set_title("Math score")
sns.barplot(data = data, x='gender', y='reading score',edgecolor='#0000',**{'alpha':0.8,'linewidth':2},ax = axes[1],ci=None)
axes[1].set_title("Reading score")
sns.barplot(data = data, x='gender', y='writing score',edgecolor='#0000',**{'alpha':0.8,'linewidth':2},ax = axes[2],ci=None)
axes[2].set_title("Writing score")
sns.barplot(data = data, x='gender', y='Percentage',edgecolor='#0000',**{'alpha':0.8,'linewidth':2},ax = axes[3],ci=None)
axes[3].set_title("Total score")

for i in range(0,4):
    axes[i].set_ylim(55,75)
plt.show()
```



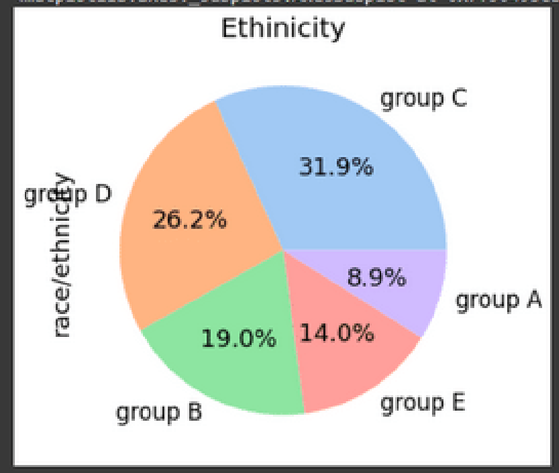
FACTOR RACE/ETHNICITY

```
[ ]  #show count Race
     data['race/ethnicity'].value_counts()
```

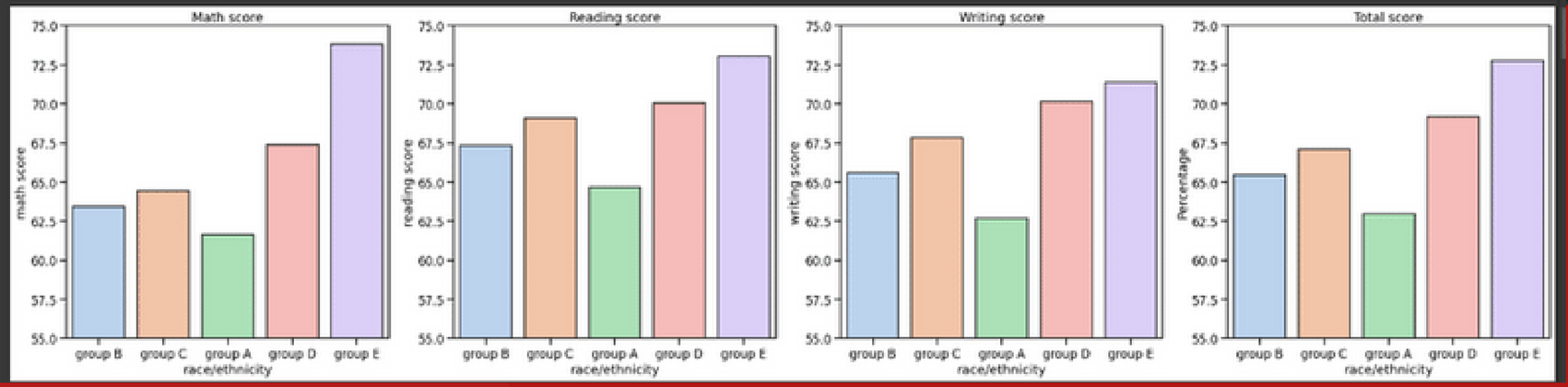
```
group C    319
group D    262
group B    190
group E    140
group A     89
Name: race/ethnicity, dtype: int64
```

```
plt.figure(figsize=(30,20))
plt.subplots_adjust(left=0.125, bottom=0.1, right=0.9, top=0.9,
                    wspace=0.5, hspace=0.2)

plt.subplot(142)
plt.title('Ethinicity',fontsize = 20)
data['race/ethnicity'].value_counts().plot.pie(autopct="%1.1f%%")
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f90495cb410>
```




```
fig , axes = plt.subplots(1,4, figsize=(37,8))
sns.barplot(data = data, x='race/ethnicity', y='math score',edgecolor='#0000',**{'alpha':0.8,'linewidth':2},ax = axes[0],ci=None)
axes[0].set_title("Math score")
sns.barplot(data = data, x='race/ethnicity', y='reading score',edgecolor='#0000',**{'alpha':0.8,'linewidth':2},ax = axes[1],ci=None)
axes[1].set_title("Reading score")
sns.barplot(data = data, x='race/ethnicity', y='writing score',edgecolor='#0000',**{'alpha':0.8,'linewidth':2},ax = axes[2],ci=None)
axes[2].set_title("Writing score")
sns.barplot(data = data, x='race/ethnicity', y='Percentage',edgecolor='#0000',**{'alpha':0.8,'linewidth':2},ax = axes[3],ci=None)
axes[3].set_title("Total score")

for i in range(0,4):
    axes[i].set_ylim(55,75)
plt.show()
```

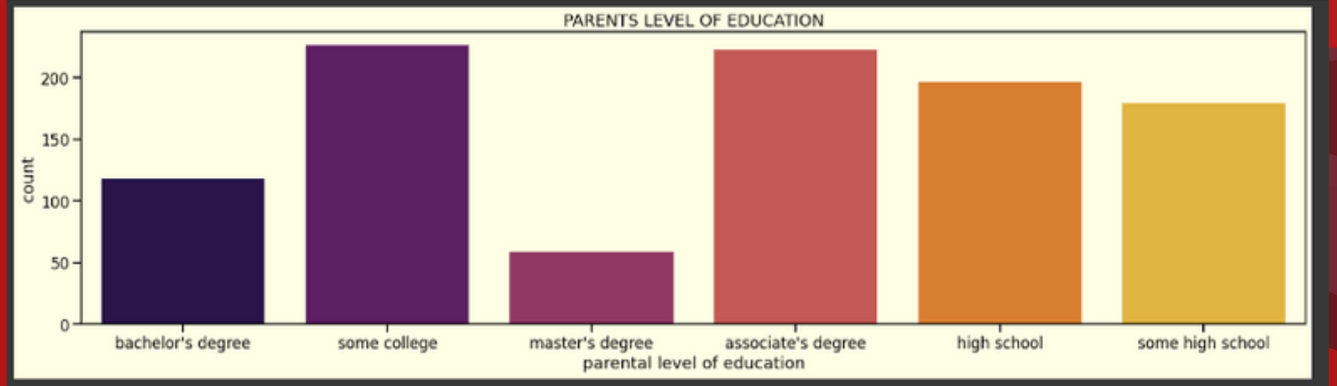


FACTOR PARENTAL LEVEL OF EDUCATION

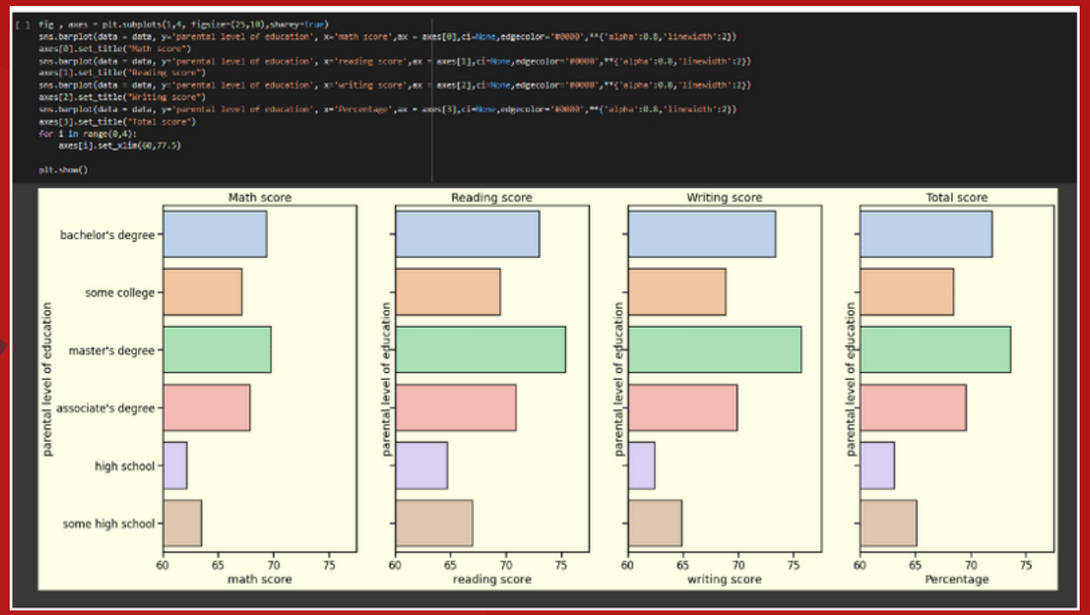
```
#show count Race
data['parental level of education'].value_counts()
```

```
some college          226
associate's degree    222
high school           196
some high school      179
bachelor's degree     118
master's degree        59
Name: parental level of education, dtype: int64
```

```
plt.rcParams['figure.facecolor'] = "#ffffa6"
plt.rcParams['axes.facecolor'] = "#ffffe6"
plt.figure(figsize=(20,6))
plt.title('PARENTS LEVEL OF EDUCATION')
sns.countplot(x='parental level of education',data=data,palette='inferno')
plt.tight_layout()
```




```
fig , axes = plt.subplots(1,4, figsize=(25,10),sharey=True)
sns.barplot(data = data, y='parental level of education', x='math score',ax = axes[0],ci=None,edgecolor='#0000',**{'alpha':0.8,'linewidth':2})
axes[0].set_title("Math score")
sns.barplot(data = data, y='parental level of education', x='reading score',ax = axes[1],ci=None,edgecolor='#0000',**{'alpha':0.8,'linewidth':2})
axes[1].set_title("Reading score")
sns.barplot(data = data, y='parental level of education', x='writing score',ax = axes[2],ci=None,edgecolor='#0000',**{'alpha':0.8,'linewidth':2})
axes[2].set_title("Writing score")
sns.barplot(data = data, y='parental level of education', x='Percentage',ax = axes[3],ci=None,edgecolor='#0000',**{'alpha':0.8,'linewidth':2})
axes[3].set_title("Total score")
for i in range(0,4):
    axes[i].set_xlim(60,77.5)

plt.show()
```

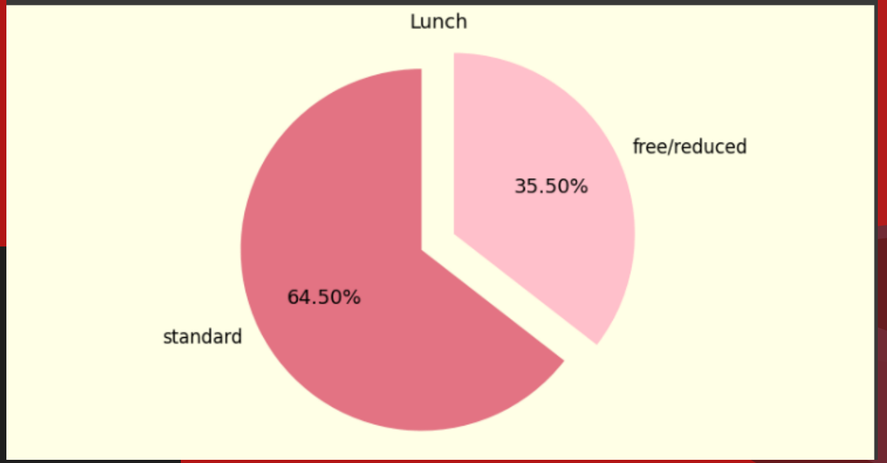


FACTOR LUNCH

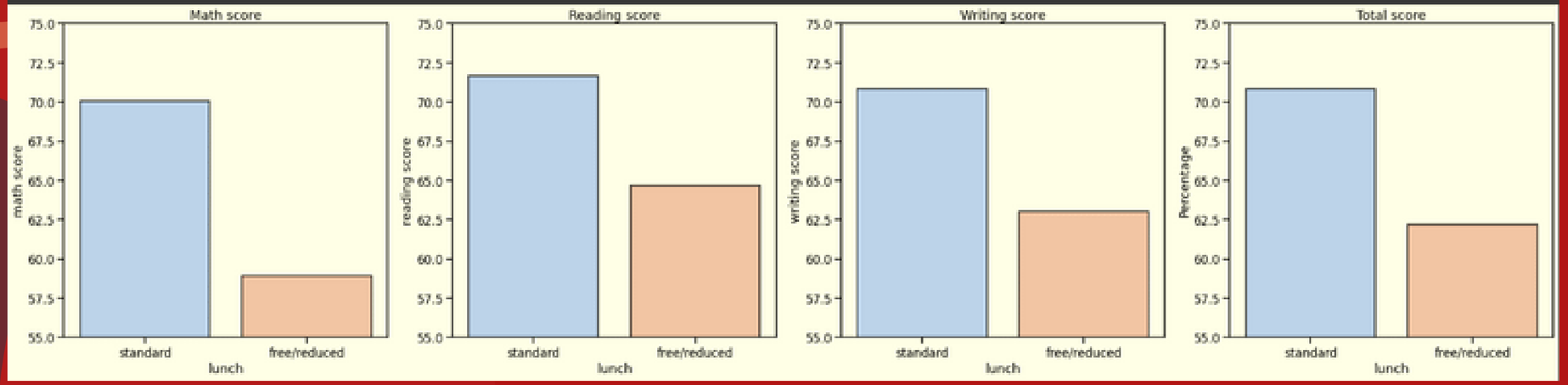
```
1 #show count Race
2 data['lunch'].value_counts()
```

```
standard        645
free/reduced    355
Name: lunch, dtype: int64
```

```
1 plt.figure(figsize=(14, 7))
2 labels=['standard', 'free/reduced']
3 plt.pie(data['lunch'].value_counts(), labels=labels, explode=[0.1,0.1],
4         autopct='%1.2f%%',colors=['#E37383','#FFC0CB'], startangle=90)
5 plt.title('Lunch')
6 plt.axis('equal')
7 plt.show()
```




```
1 fig , axes = plt.subplots(1,4, figsize=(37,8))
2 sns.barplot(data = data, x='lunch', y='math score',edgecolor='#0000',**{'alpha':0.8,'linewidth':2},ax = axes[0],ci=None)
3 axes[0].set_title("Math score")
4 sns.barplot(data = data, x='lunch', y='reading score',edgecolor='#0000',**{'alpha':0.8,'linewidth':2},ax = axes[1],ci=None)
5 axes[1].set_title("Reading score")
6 sns.barplot(data = data, x='lunch', y='writing score',edgecolor='#0000',**{'alpha':0.8,'linewidth':2},ax = axes[2],ci=None)
7 axes[2].set_title("Writing score")
8 sns.barplot(data = data, x='lunch', y='Percentage',edgecolor='#0000',**{'alpha':0.8,'linewidth':2},ax = axes[3],ci=None)
9 axes[3].set_title("Total score")
10
11 for i in range(0,4):
12     axes[i].set_ylim(55,75)
13 plt.show()
```

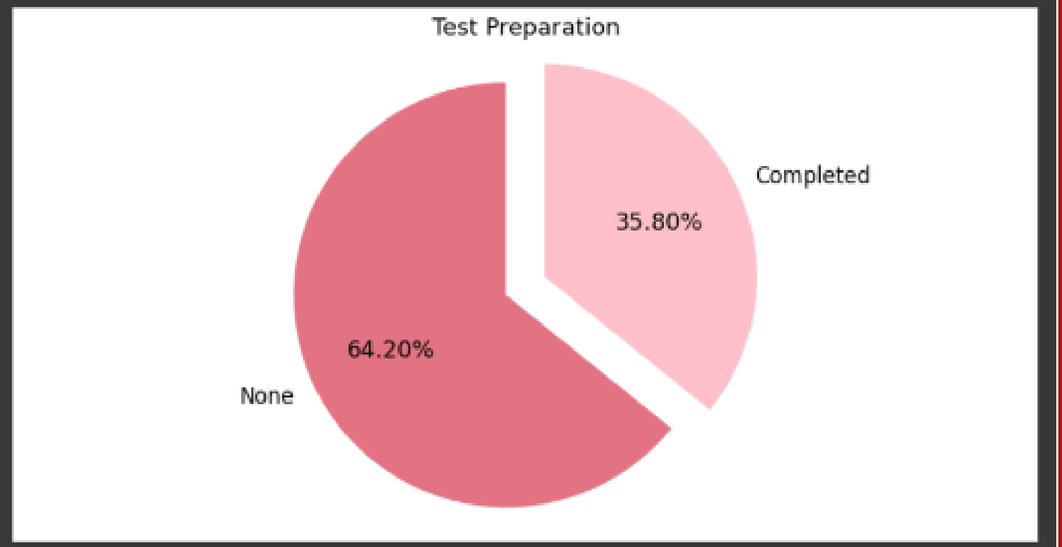


FACTOR TEST PREPARATION COURSE

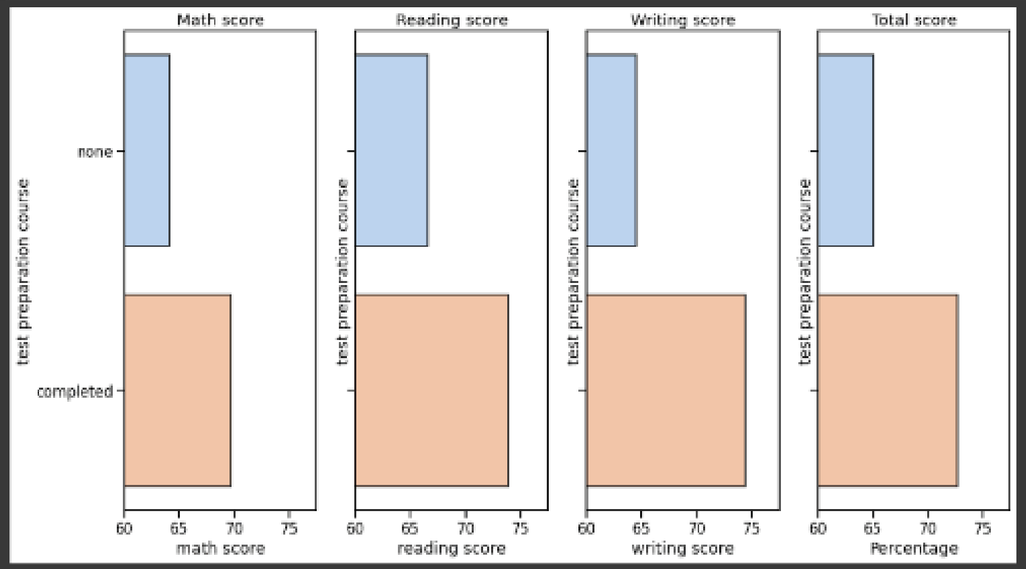
```
#show count Race
data['test preparation course'].value_counts()
```

```
none         642
completed    358
Name: test preparation course, dtype: int64
```

```
plt.figure(figsize=(14, 7))
labels=['None', 'Completed']
plt.pie(data['test preparation course'].value_counts(), labels=labels, explode=[0.1,0.1],
        autopct='%1.2f%%',colors=['#E37383','#FFC0CB'], startangle=90)
plt.title('Test Preparation')
plt.axis('equal')
plt.show()
```




```
fig , axes = plt.subplots(1,4, figsize=(18,10),sharey=True)
sns.barplot(data = data, y='test preparation course', x='math score',ax = axes[0],ci=None,edgecolor='#0000',**{'alpha':0.8,'linewidth':2})
axes[0].set_title("Math score")
sns.barplot(data = data, y='test preparation course', x='reading score',ax = axes[1],ci=None,edgecolor='#0000',**{'alpha':0.8,'linewidth':2})
axes[1].set_title("Reading score")
sns.barplot(data = data, y='test preparation course', x='writing score',ax = axes[2],ci=None,edgecolor='#0000',**{'alpha':0.8,'linewidth':2})
axes[2].set_title("Writing score")
sns.barplot(data = data, y='test preparation course', x='Percentage',ax = axes[3],ci=None,edgecolor='#0000',**{'alpha':0.8,'linewidth':2})
axes[3].set_title("Total score")
for i in range(0,4):
    axes[i].set_xlim(60,77.5)

plt.show()
```



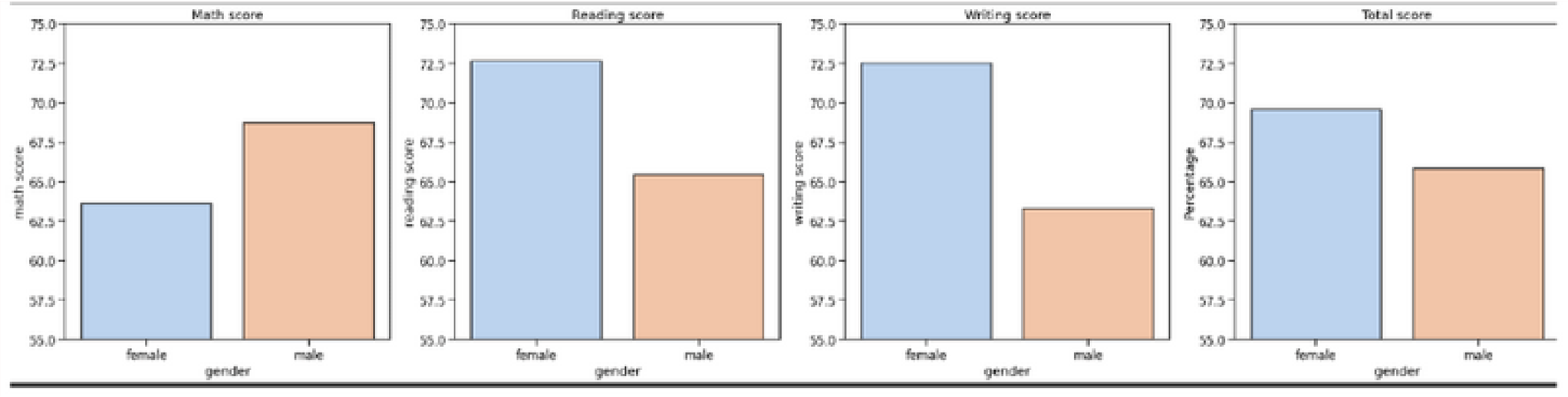
DISCUSSION

Result

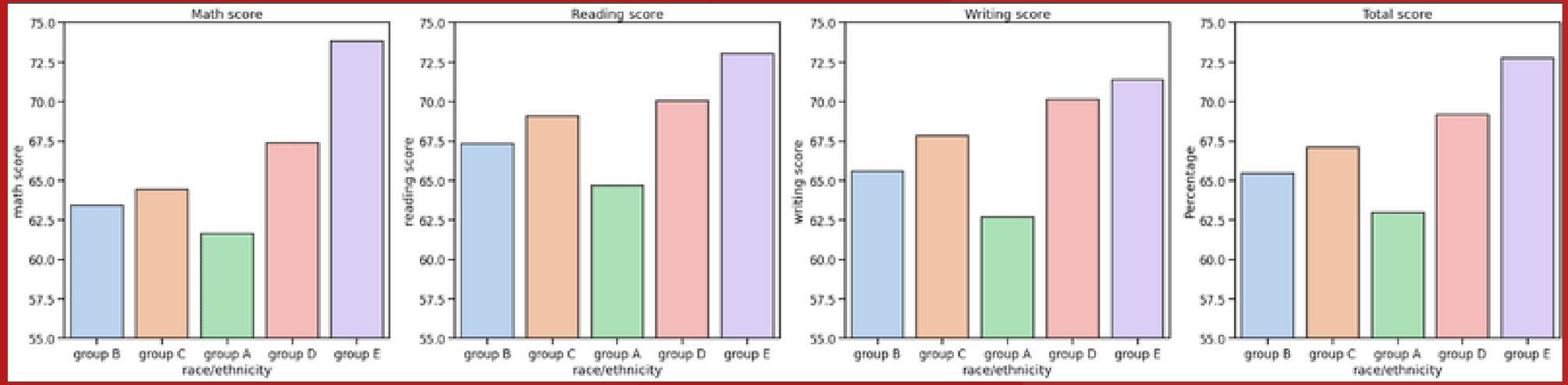
Rumusan Masalah 1

Setelah menganalisis dataset kami, kami dapat membuktikan bahwa score exam siswa dapat dipengaruhi oleh semua faktor yang terdapat di dataset, karena setiap faktor akan menghasilkan hasil score berbeda-beda.

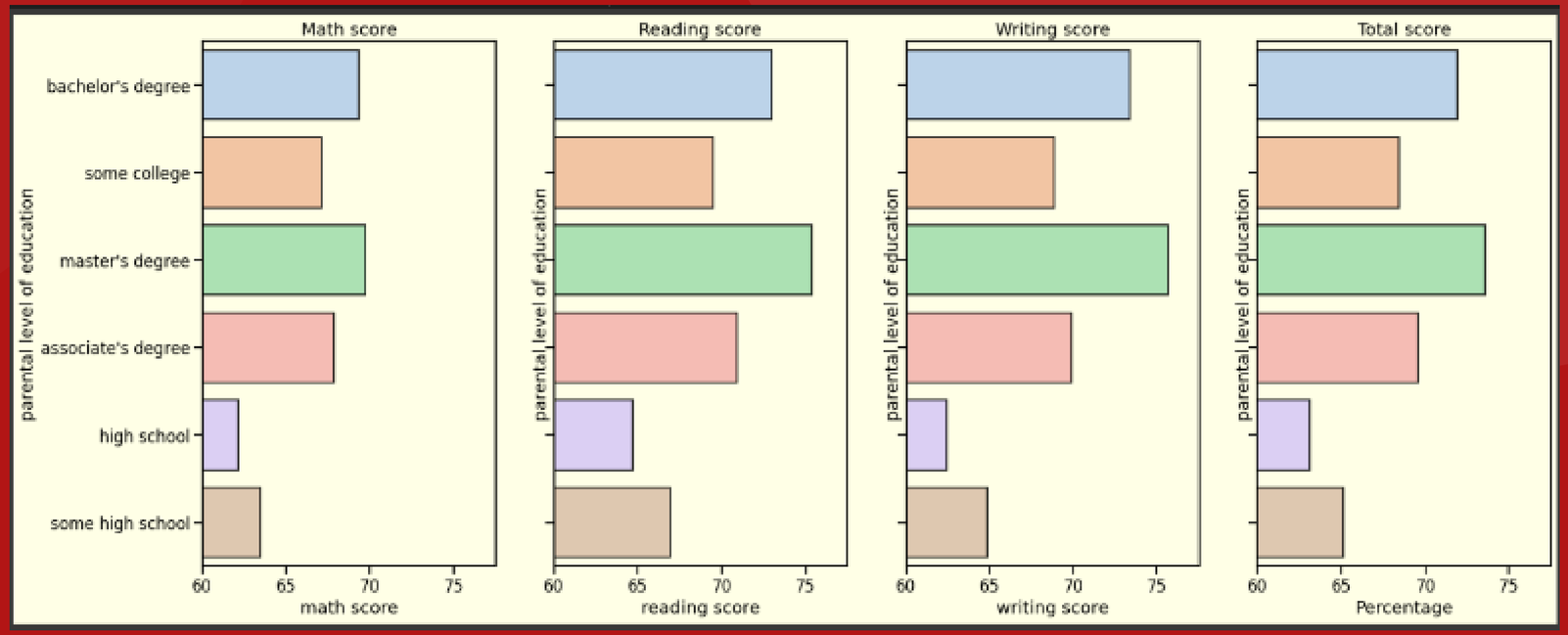
- ## FACTOR GENDER



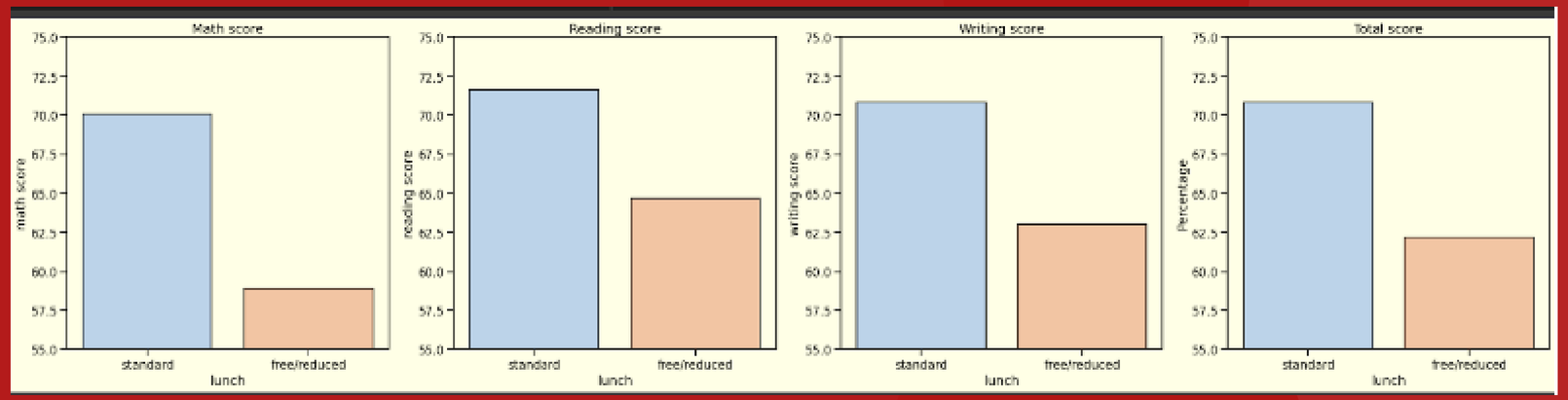
- FACTOR RACE/ETHNICITY



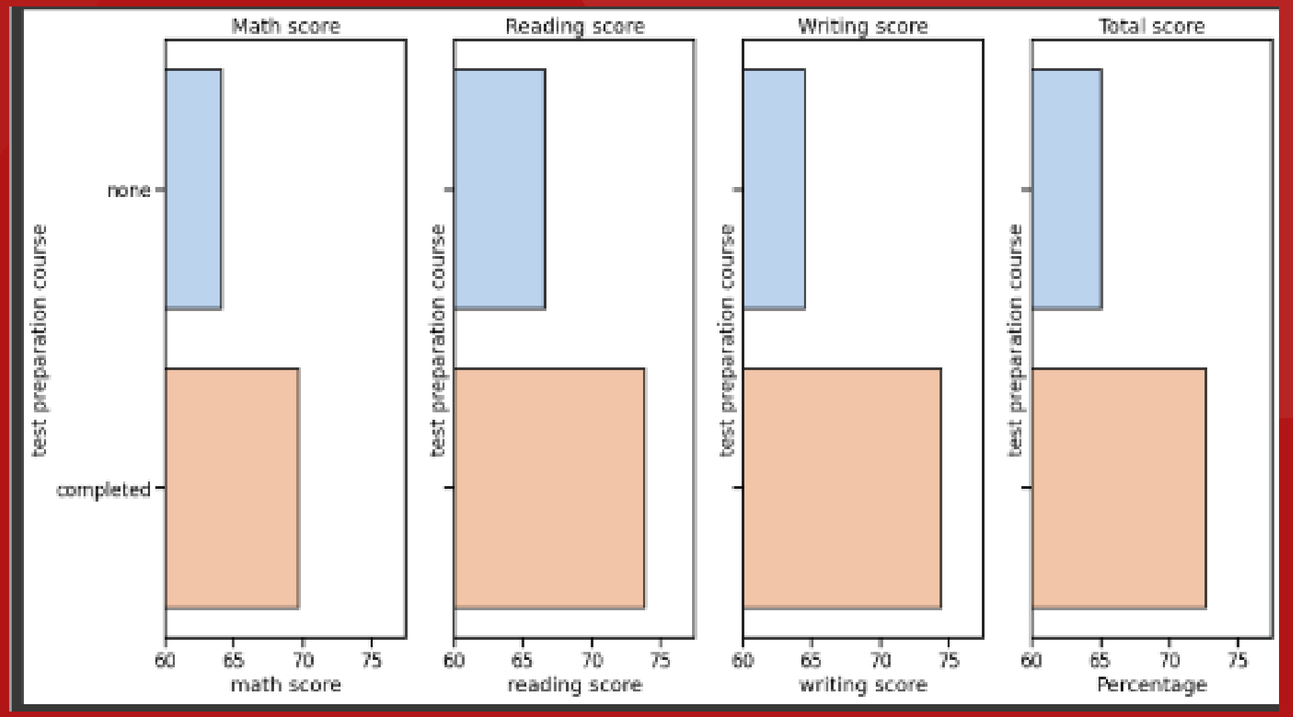
- FACTOR PARENTAL LEVEL OF EDUCATION



- FACTOR LUNCH

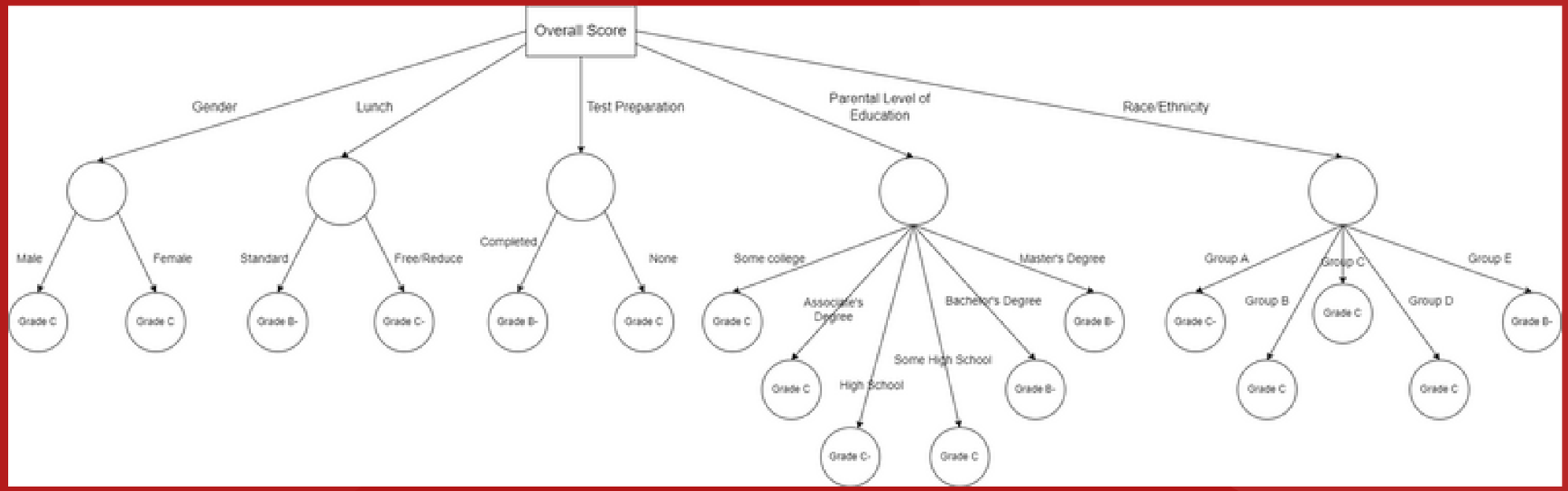


- FACTOR TEST PREPARATION COURSE



Rumusan Masalah 2:

Setelah kami memproses data yang ada, kami membuat sebuah decision tree yang berguna untuk mengetahui faktor yang paling berpengaruh terhadap performa siswa.



Tingkat kepengaruhan:

- Tidak berpengaruh: kondisi saat perbandingan grade sama yaitu 1:1
- Berpengaruh: kondisi saat perbandingan grade 1:2
- Sangat Berpengaruh: kondisi saat grade memiliki perbandingan >= 1:3

THANK YOU
AND SEE YOU AGAIN

BINA NUSANTARA UNIVERSITY