```
import pandas as pd

def calculate_demographic_data(print_data=True):
    # Read data from file
    df = pd.read_csv("adult.data.csv")

    # 1. How many people of each race are represented in this dataset?
    race_count = df['race'].value_counts()

    # 2. What is the average age of men?
    average_age_men = round(df[df['sex'] == 'Male']['age'].mean(), 1)

    # 3. What is the percentage of people who have a Bachelor's degree?
    total_people = len(df)
    bachelors_count = len(df[df['education'] == 'Bachelors'])
    percentage_bachelors = round((bachelors_count / total_people) * 100, 1)

    # 4. % with advanced education making >50K
    higher_education = df[df['education'].isin(['Bachelors', 'Masters', 'Doctorate'])]
    lower_education = df[~df['education'].isin(['Bachelors', 'Masters', 'Doctorate'])]

    higher_education_rich = round(
        (len(higher_education[higher_education['salary'] == '>50K']) / len(higher_education)) *
100, 1
    )
```

```
lower_education_rich = round(
    (len(lower_education[lower_education['salary'] == '>50K']) / len(lower_education)) * 100, 1
)

# 5. What is the minimum number of hours a person works per week?
min_work_hours = df['hours-per-week'].min()

# 6. % of people working min hours and earning >50K
min_workers = df[df['hours-per-week'] == min_work_hours]
rich_min_workers = min_workers[min_workers['salary'] == '>50K']
rich_percentage = round((len(rich_min_workers) / len(min_workers)) * 100, 1)

# 7. Country with highest % of people earning >50K
country_counts = df['native-country'].value_counts()
rich_country_counts = df[df['salary'] == '>50K']['native-country'].value_counts()
rich_country_percentages = (rich_country_counts / country_counts) * 100
highest_earning_country = rich_country_percentages.idxmax()
highest_earning_country_percentage = round(rich_country_percentages.max(), 1)

# 8. Most popular occupation for those who earn >50K in India
top_IN_occupation = df[
    (df['native-country'] == 'India') & (df['salary'] == '>50K')
]['occupation'].value_counts().idxmax()

# Output dictionary
if print_data:
```

```
    print("Number of each race:\n", race_count)
    print("Average age of men:", average_age_men)
    print("Percentage with Bachelors degrees:", percentage_bachelors)
    print("Percentage with higher education that earn >50K:", higher_education_rich)
    print("Percentage without higher education that earn >50K:", lower_education_rich)
    print("Min work time:", min_work_hours)
    print("Percentage of rich among those who work fewest hours:", rich_percentage)
    print("Country with highest percentage of rich:", highest_earning_country)
    print("Highest percentage of rich people in country:",
highest_earning_country_percentage)
    print("Top occupations in India:", top_IN_occupation)

  return {
    'race_count': race_count,
    'average_age_men': average_age_men,
    'percentage_bachelors': percentage_bachelors,
    'higher_education_rich': higher_education_rich,
    'lower_education_rich': lower_education_rich,
    'min_work_hours': min_work_hours,
    'rich_percentage': rich_percentage,
    'highest_earning_country': highest_earning_country,
    'highest_earning_country_percentage': highest_earning_country_percentage,
    'top_IN_occupation': top_IN_occupation
  }
```