Task 5: Decision Trees & Random Forests - Heart Disease Classification

Objective
Implement decision tree and random forest classifiers to predict heart disease. Provide visualizations, overfitting control, and cross-validation evaluation. The repository contains a ready-to-run script and sample outputs so you can upload directly.

Files Included
- `decision_tree_random_forest.py` : Main script. Place `heart.csv` beside this script and run.
- `README.md` : This document.
- `outputs/` : (Generated by the script) cleaned dataset, plots, and report.

How to run
1. Download the Heart Disease dataset (CSV) from Kaggle or appropriate source and save it as `heart.csv` in the same directory.
2. Ensure Python 3.8+ and required packages are installed:
   ```bash
   pip install pandas numpy scikit-learn matplotlib seaborn
   ```
3. Run the script:
   ```bash
   python decision_tree_random_forest.py
   ```
4. Check the `outputs/` folder for plots and `report.txt`.

Notes for reviewers
- The script tries to detect the target column automatically (common names: `target`, `heart_disease`, etc.).
- It encodes categorical features with one-hot encoding and standardizes numeric features.
- Uses GridSearchCV to tune Decision Tree and Random Forest hyperparameters (depth, n_estimators, etc.).
- Produces: decision tree image, random forest feature importances, accuracy comparison, confusion matrix, and a short report.

Dataset Source Ideas
- UCI Heart Disease datasets or Kaggle heart disease datasets.

License
You may reuse and adapt this code for learning purposes.