matrix

```
Matrix();
Matrix(int row, int col);
Matrix(int n);
Matrix(double arr[], int n);
Matrix(const Matrix& N);
void set_value( int row, int col, double newVal);
double get_value(int row,int col) const;
void clear();
~Matrix();
int getRow() const;
int getCol() const;
friend ostream& operator<<(ostream& os, const Matrix& mat);
friend bool operator==( const Matrix& mat1, const Matrix& mat2);
friend bool operator!=( const Matrix& mat1, const Matrix& mat2);
Matrix& operator++();
Matrix operator++(int);
Matrix& operator--();
Matrix operator--(int);
Matrix& operator=(Matrix mat);
friend void mySwap(Matrix &mat1, Matrix &mat2);
friend Matrix operator+(const Matrix& mat1, const Matrix& mat2);
Matrix& operator+=(Matrix &mat);
friend Matrix operator-(const Matrix& mat1, const Matrix& mat2);
Matrix& operator-=(Matrix& mat);
friend Matrix operator*(const Matrix& mat1, const Matrix& mat2);
Matrix& operator*=(const Matrix& mat);
int col, row;
double * a;
void allocateArr();
```

main

+main()

conMatrix

+default()
+conMatrix(int)
+contMatrix(double*, int)
+conMatrixx(const conMatrix&)
+~conMatrix()
+operator <<(ostream &, const conMatrix &)

public

PageRank

+default()
+PageRank()
+createG()
+createS(const Matrix&)
+createSBlank(const Matrix&)
+createQ()
+createR(const Matrix&, const Matrix&)
+createRank(const Matrix&)
+multiplyRank(const Matrix&, const Matrix...
+divideRank(const Matrix&)
+printResults()
int W;
double *array;
int sum;