General Certificate of Education
Advanced Subsidiary Examination
June 2015

Computing COMP1/PM

Unit 1 Problem Solving, Programming, Data Representation and Practical Exercise

Preliminary Material

To be issued to candidates on or after **Sunday 1 March 2015**, subject to the instructions given in the Teachers' Notes (COMP1/TN).

Information

- The Preliminary Material is to be seen by teachers and candidates **only**, for use during preparation for the examination on Monday 1 June 2015. It **cannot** be used by anyone else for any other purpose, other than as stated in the instructions issued, until after the examination date has passed. It must **not** be provided to third parties.
- This Preliminary Material comprises Instructions for Candidates.
- A Skeleton Program is provided separately by your teacher and must be read in conjunction with this Preliminary Material.
- Candidates are advised to familiarise themselves with the Preliminary Material and Skeleton Program before the examination.
- Another copy of this Preliminary Material will be made available to you in the examination. You will also be given access to the Skeleton Program electronically at the start of the examination. You must **not** take any copy of the Preliminary Material, Skeleton Program or any other material into the examination room.

INSTRUCTIONS FOR CANDIDATES

The question paper is divided into four sections and a recommendation is given to candidates as to how long to spend on each section. Below are the recommended timings for the 2015 examination.

SECTION A
You are advised to spend no more than **35 minutes** on this section.
Questions will examine the specification content **not** specific to the **Preliminary Material.**

SECTION B
You are advised to spend no more than **15 minutes** on this section.
You will be asked to create a new program **not** related to the **Preliminary Material** or **Skeleton Program**.

SECTION C
You are advised to spend no more than **10 minutes** on this section.
Questions will refer to the **Preliminary Material** and the **Skeleton Program**, but will not require programming.

SECTION D
You are advised to spend no more than **60 minutes** on this section.
Questions will use the **Skeleton Program** and the **Preliminary Material**.

Electronic Answer Document

Answers for all questions for all sections must be entered into the word processed document made available to the candidate at the start of the examination and referred to in the question paper rubrics as the **Electronic Answer Document**.

Preparation for the Examination

For your programming language you should ensure that you are familiar with this **Preliminary Material** and the **Skeleton Program**.

For the **Skeleton Program** for your programming language, you should be familiar with the built-in functions available for manipulating string data.

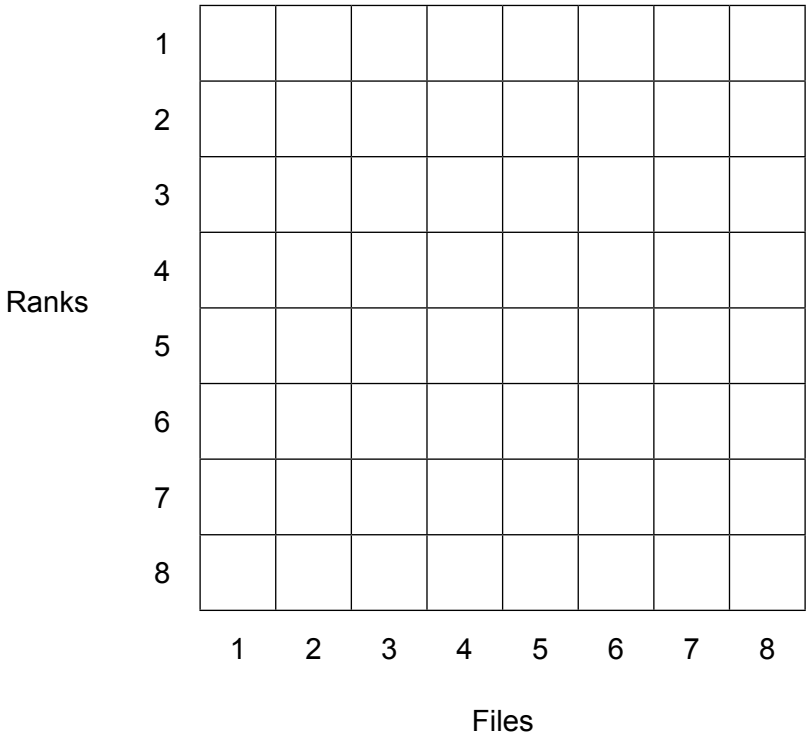
CAPTURE THE SARRUM

The **Skeleton Program** in this Preliminary Material is for the board game CAPTURE THE SARRUM.

CAPTURE THE SARRUM is an ancient variant of the game of chess. It is a two-player game – one player has the white pieces and the other player has the black pieces. Each player takes it in turn to move one of their pieces; different pieces have different rules that determine how they can move. To win the game, a player needs to take (capture) the opponent's sarrum. The player using the white pieces always moves first.

Figure 1 shows an empty board. The board consists of an 8x8 grid of squares. The rows are called **ranks** and the columns are called **files**. Each file is labelled with a number between one and eight and each rank is also labelled with a number between one and eight.

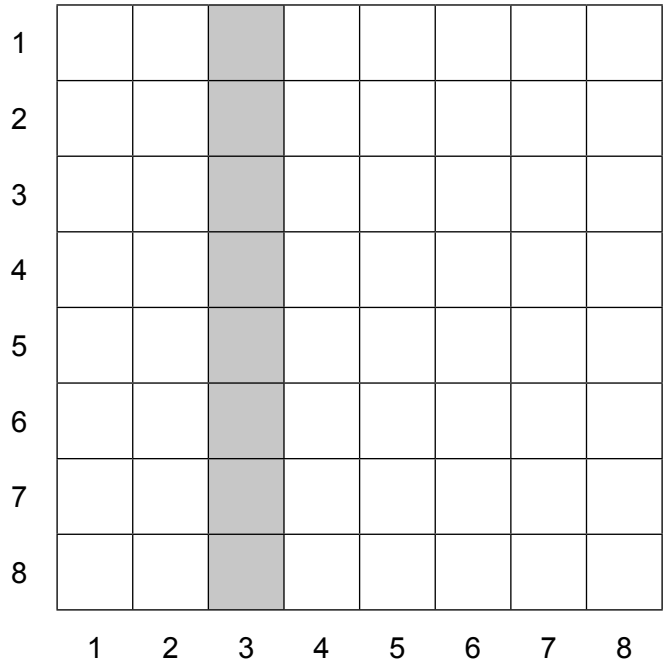
Figure 1



The preliminary material continues on the next page

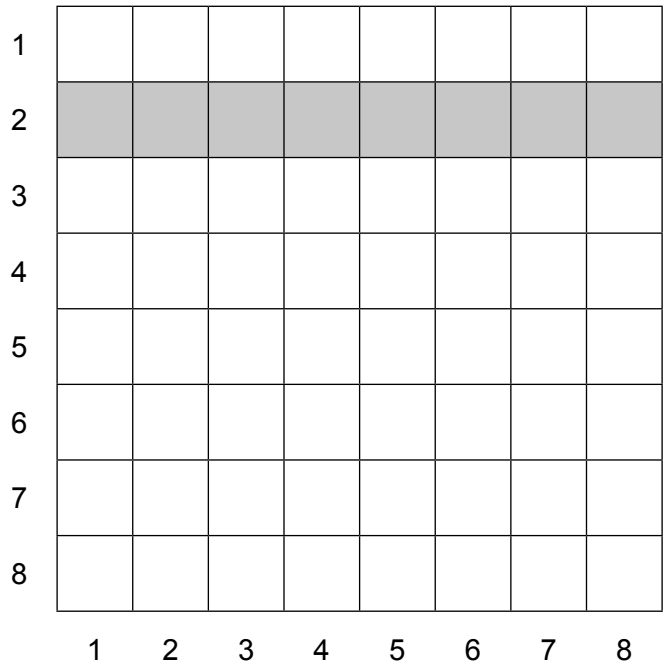
In **Figure 2** the third file has been shaded.

Figure 2



In **Figure 3** the second rank has been shaded.

Figure 3



There are six different types of piece in the game; the rules for moving each of the pieces are described in **Table 1**. If a piece moves into a square containing one of the opponent's pieces, then that piece is taken (it is removed from the board). No piece can move into a square containing another piece of the same colour.

Table 1

Piece name	**How the piece moves**
Sarrum (king)	A sarrum can move one square at a time in any direction.
Marzaz pani (royal attendant)	A marzaz pani can move one square at a time either vertically or horizontally – it cannot make diagonal moves.
Nabu (seer)	A nabu can move one square at a time along a diagonal – it cannot make horizontal or vertical moves.
Etlu (warrior)	An etlu can move two squares at a time in either a vertical or horizontal line – it must move exactly two squares and it cannot make diagonal moves. An etlu can jump over other pieces.
Gisgigir (chariot)	A gisgigir can move any number of squares at a time in either a vertical or horizontal line – it cannot make diagonal moves. A gisgigir cannot jump over other pieces.
Redum (soldier)	A redum can move forward one square at a time. If the square directly in front of a redum is empty then the redum can move into that square. If a square that is one rank in front of and directly to the left or right of a redum contains an opponent's piece, then the redum can move into that square and will capture the piece. If a redum reaches the back rank (the first rank for a white redum, the eighth rank for a black redum), it is promoted (changes) into a marzaz pani.

Historical records for the game of CAPTURE THE SARRUM are incomplete – some records indicate that there may have been another type of piece called a kashshaptu (witch) but it is not known how this piece was used in the game.

The preliminary material continues on the next page

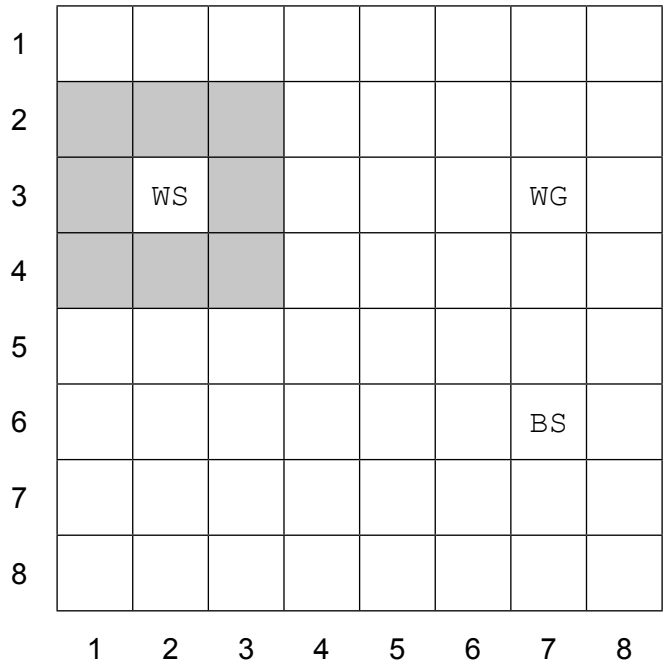
Figure 4 shows the position of the pieces at the start of a game of CAPTURE THE SARRUM. Each piece has been represented using the first letter of its name, eg `S` = sarrum. A `B` is used to indicate a black piece and a `W` indicates a white piece.

Figure 4

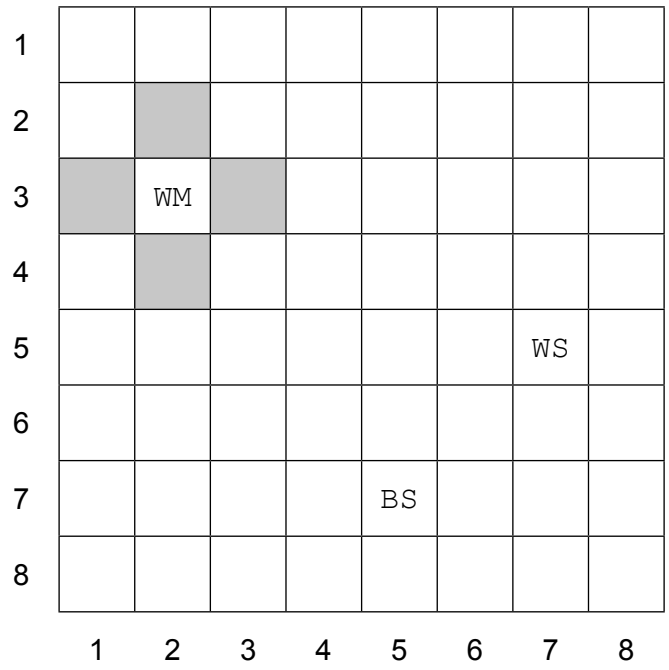
	1	2	3	4	5	6	7	8
1	BG	BE	BN	BM	BS	BN	BE	BG
2	BR	BR	BR	BR	BR	BR	BR	BR
3								
4								
5								
6								
7	WR	WR	WR	WR	WR	WR	WR	WR
8	WG	WE	WN	WM	WS	WN	WE	WG

In **Figure 5** shaded squares in each diagram show the legal moves that a piece can make.

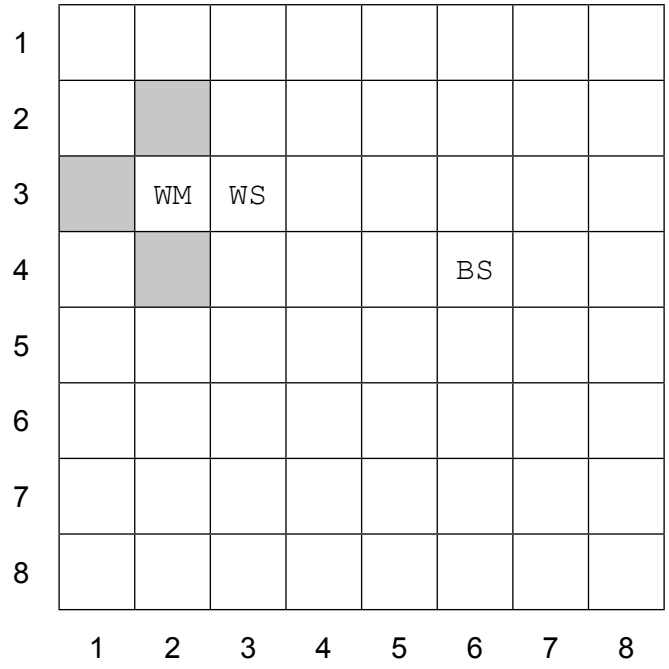
Figure 5



White's sarrum **can** move one square in any direction.

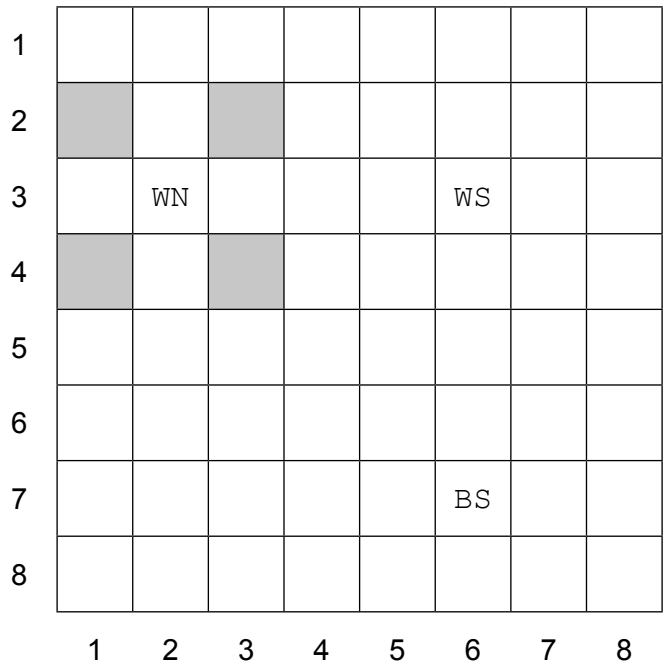


White's marzaz pani **can** move one square vertically or horizontally.

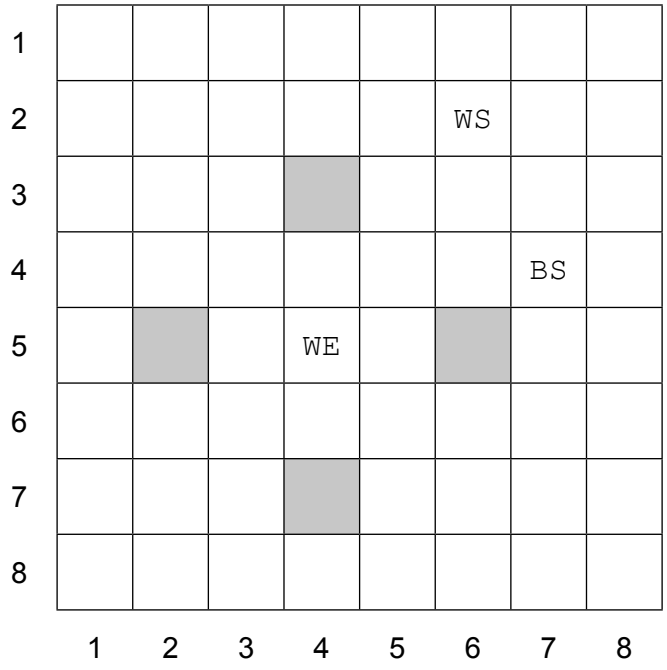


White's marzaz pani **cannot** move to a square containing another white piece.

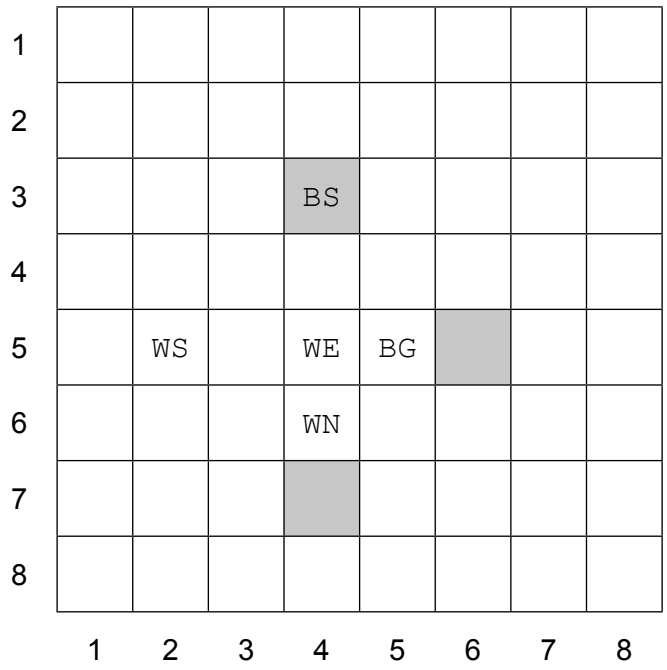
The preliminary material continues on the next page



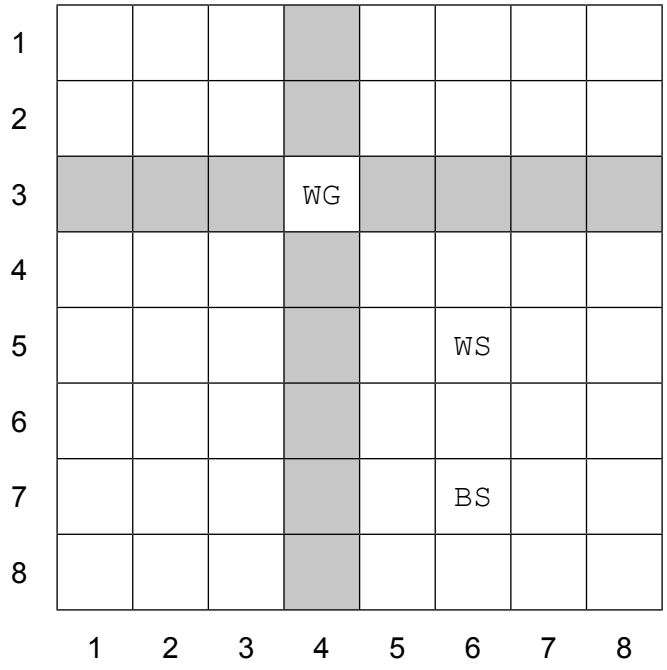
White's nabu **can** move one square in a diagonal direction.



White's etlu **can** move two squares in a vertical or horizontal direction.

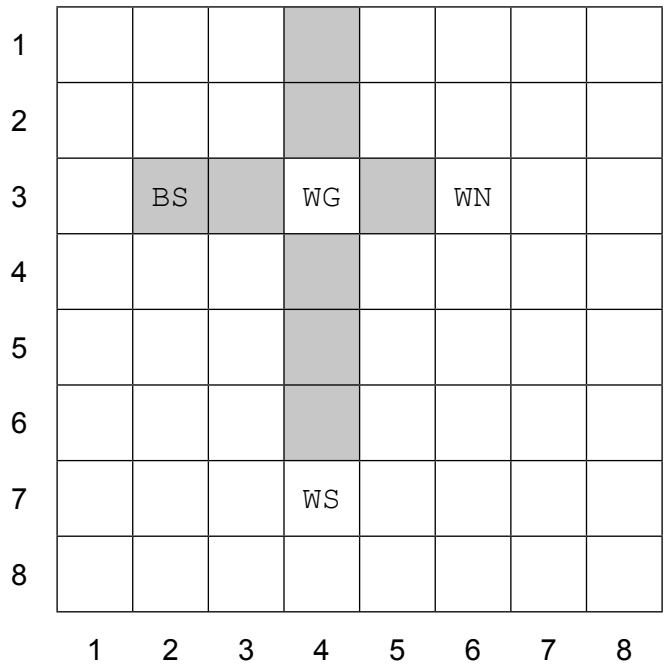


White's etlu **cannot** move into the same square as another white piece but **can** move into the same square as a black piece (the black piece will be captured). An etlu **can** jump over other pieces of either colour.

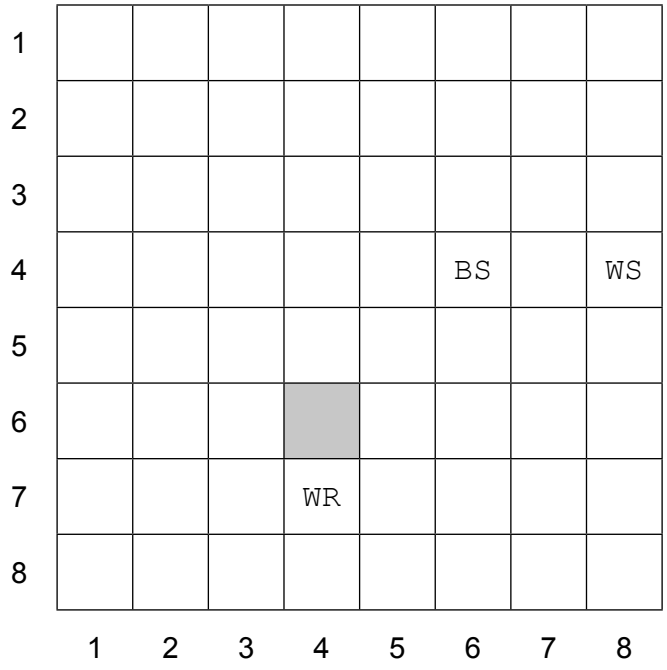


White's gisgigir **can** move any number of squares in a straight line – either horizontally or vertically.

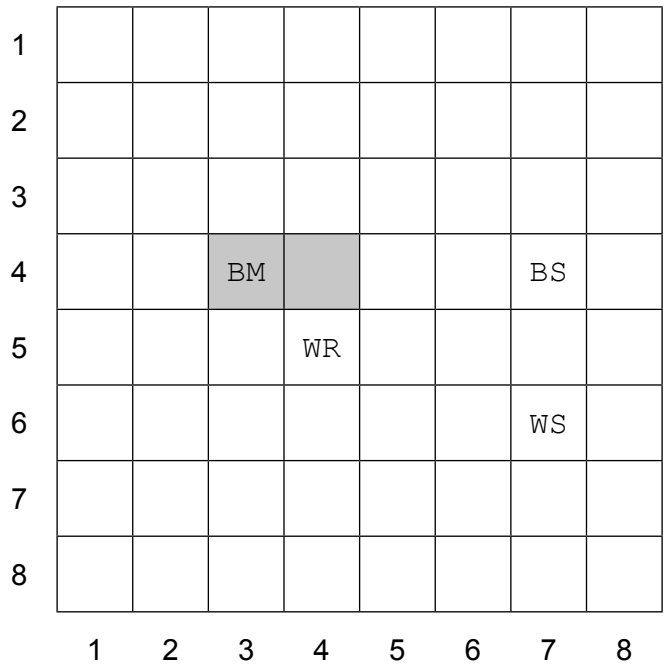
The preliminary material continues on the next page



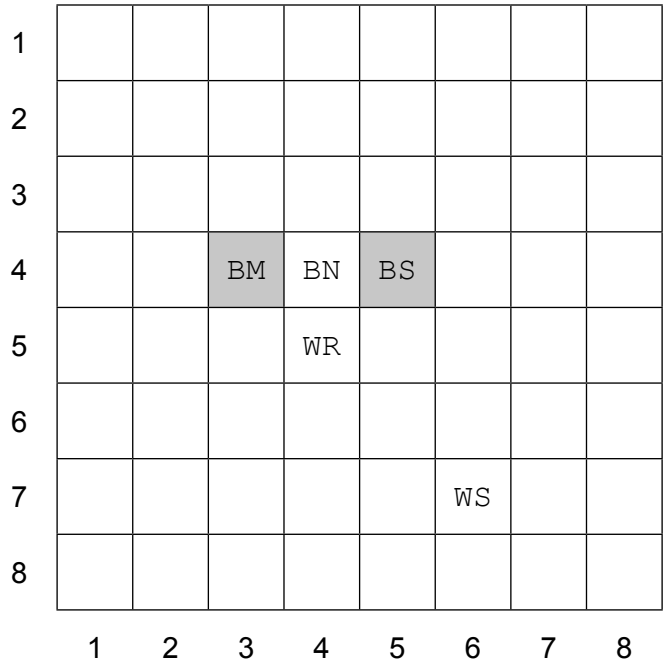
White's gisgigir **cannot** move to a square occupied by another white piece and it **cannot** jump over other pieces.



White's redum **can** move one square forward, it **cannot** move diagonally forward, unless it is capturing a black piece.

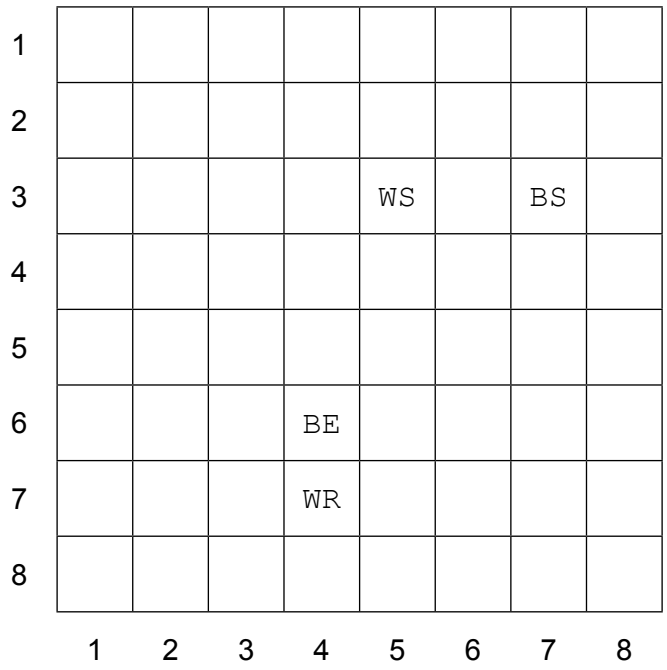


White's redum **can** either move one square forward or it **can** move diagonally forward one square to capture the black marzaz pani.

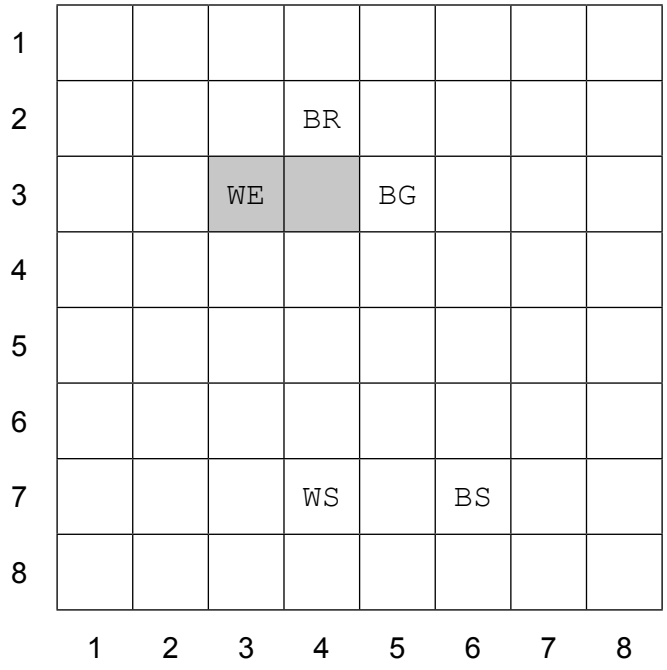


White's redum **cannot** move to the square immediately in front of it, as this is occupied by another piece; it **can** move diagonally forward one square to capture either the black marzaz pani or the black sarrum.

The preliminary material continues on the next page



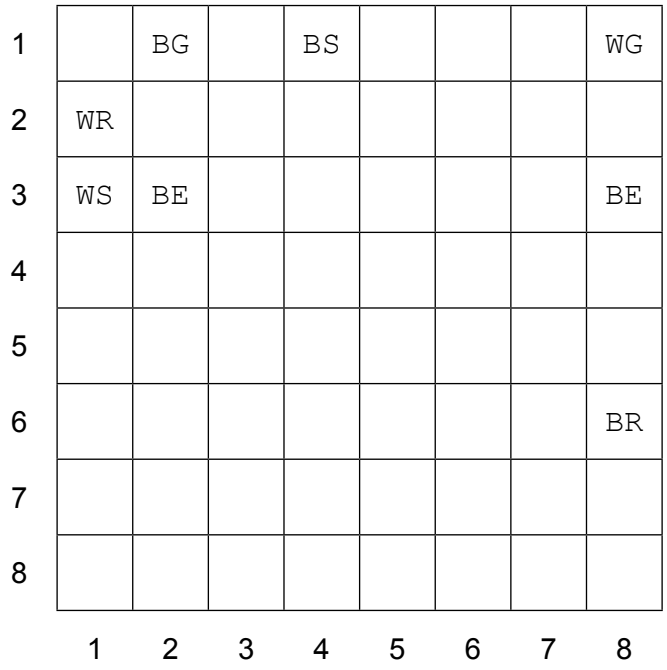
White's redum has no legal moves.



Black's redum **can** either move one square forward or it can capture the white etlu – it **cannot** move to the square containing the black gisgigir.

When the **Skeleton Program** is run, the user is given the option to play, using a sample game. If they choose to play the sample game, then the initial board position will be as shown in **Figure 6**, otherwise the initial board position will be as shown in **Figure 4**.

Figure 6



When the sample game starts, it is White's move.

To enter a move, the player types in the grid reference of the square containing the piece that they would like to move (the start square) and then the grid reference of the square that they would like to move the selected piece to (the finish square). A grid reference is the number of the file followed by the number of the rank, eg in **Figure 6** the grid reference 41 indicates the position of Black's sarrum. If the move entered is a legal move, then the board will be updated and it will now be the other player's turn.

The preliminary material continues on the next page

Figure 7, **Figure 8** and **Figure 9** show three examples of White entering a move and the positions after these moves, from the start position shown in **Figure 6**.

Figure 7

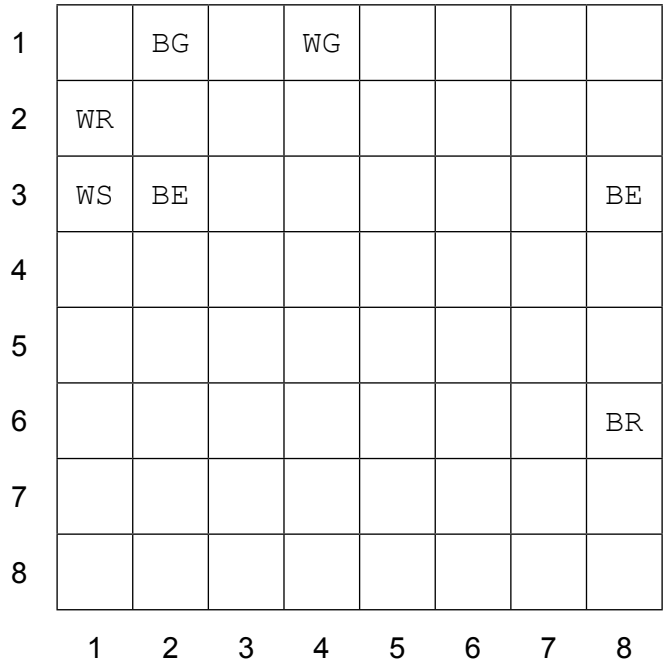
White enters a start square of 13 and a finish square of 23.

1		BG		BS				WG
2	WR							
3		WS						BE
4								
5								
6								BR
7								
8								
	1	2	3	4	5	6	7	8

White's sarrum has captured Black's etlu.

Figure 8

White enters a start square of 81 and a finish square of 41.

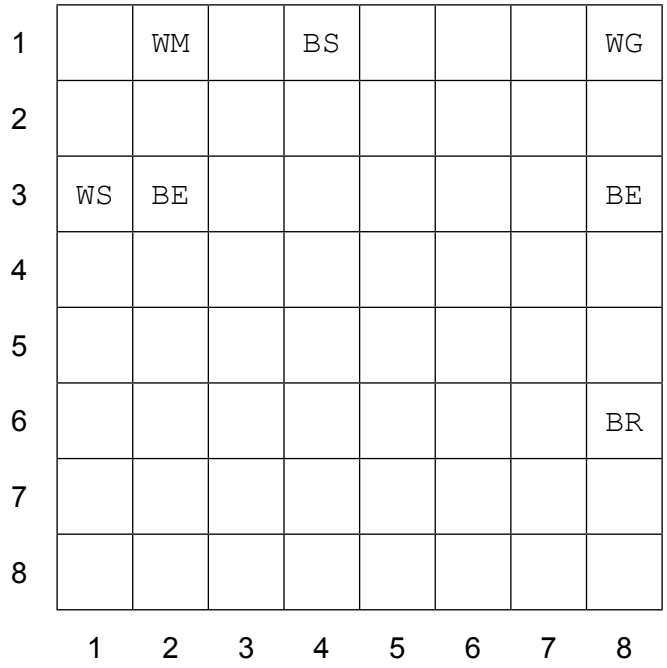


White's gisgigir has captured Black's sarrum and wins the game.

The preliminary material continues on the next page

Figure 9

White enters a start square of 12 and a finish square of 13.
A message saying “That is not a legal move – please try again” is displayed.
White then enters a start square of 12 and a finish square of 21.



White’s redum has captured Black’s gisgigir and White’s redum has reached the first rank and so is promoted to a marzaz pani.

The game continues until one of the players captures the opponent’s sarrum. The user is then given the option to play another game – if they don’t choose to play another game, then this will exit the **Skeleton Program**.

Notes

Your chosen programming language may use arrays with a lower bound value of `0`. If so, array cells with indices of `0` are not used.

END OF PRELIMINARY MATERIAL