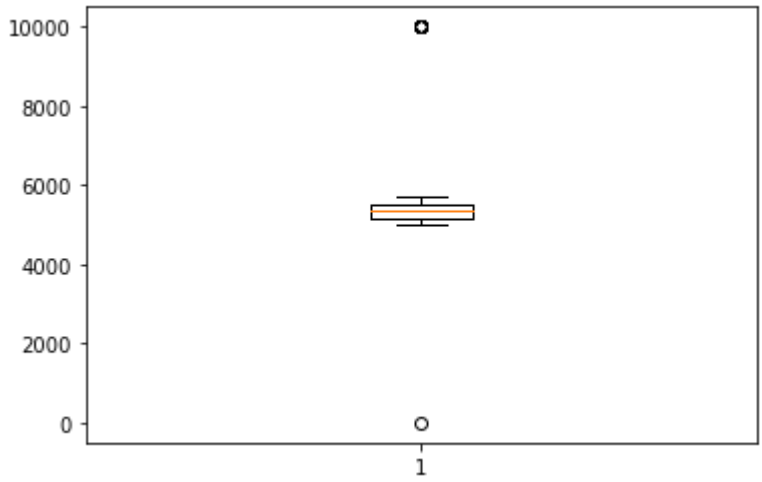
In [175]:
```
#import packages

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [176]:
```
# Import data from .csv file into dataframes

ability_ids = pd.read_csv("ability_ids.csv")
ability_upgrades = pd.read_csv("ability_upgrades.csv")
chat = pd.read_csv("chat.csv")
cluster_regions = pd.read_csv("cluster_regions.csv")
hero_names = pd.read_csv("hero_names.csv")
item_ids = pd.read_csv("item_ids.csv")
match = pd.read_csv("match.csv")
objectives = pd.read_csv("objectives.csv")
patch_dates = pd.read_csv("patch_dates.csv")
player_ratings = pd.read_csv("player_ratings.csv")
player_time = pd.read_csv("player_time.csv")
players = pd.read_csv("players.csv")
purchase_log = pd.read_csv("purchase_log.csv")
teamfights = pd.read_csv("teamfights.csv")
teamfights_players = pd.read_csv("teamfights_players.csv")
test_labels = pd.read_csv("test_labels.csv")
test_player = pd.read_csv("test_player.csv")
```

In [177]:
```
# Exploratory analysis using summary stats + boxplots

#ability ids dataset

ability_ids[:5]

ability_ids_feature_1 = ability_ids["ability_id"]
ability_ids_feature_2 = ability_ids["ability_name"]

ability_ids_feature_1.describe()
```

Out[177]:
```
count      688.000000
mean      5413.681686
std        649.786781
min          0.000000
25%       5175.750000
50%       5353.500000
75%       5527.250000
max      10002.000000
Name: ability_id, dtype: float64
```

In [234]:
```
y = list(ability_ids.ability_id)
plt.boxplot(y)
plt.show()
```



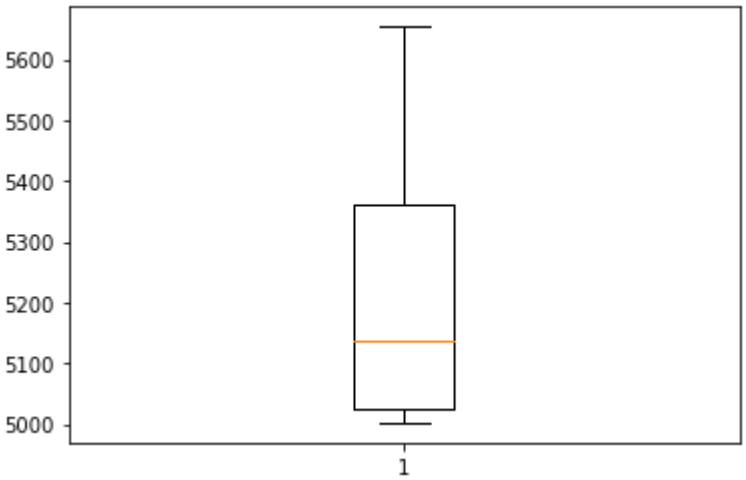

In [169]:
```
ability_ids_feature_2.describe()
```

Out[169]:
```
count              688
unique             688
top       ability_base
freq                 1
Name: ability_name, dtype: object
```

In [170]:
```
# ability_upgrades dataset

ability_upgrades[:5]

ability_upgrades_feature_1 = ability_upgrades["ability"]
ability_upgrades_feature_2 = ability_upgrades["level"]
ability_upgrades_feature_3 = ability_upgrades["time"]
ability_upgrades_feature_4 = ability_upgrades["player_slot"]
ability_upgrades_feature_5 = ability_upgrades["match_id"]

ability_upgrades_feature_1.describe()
```

Out[170]:
```
count    8.939599e+06
mean     5.204870e+03
std      1.949415e+02
min      5.002000e+03
25%      5.025000e+03
50%      5.136000e+03
75%      5.361000e+03
max      5.654000e+03
Name: ability, dtype: float64
```

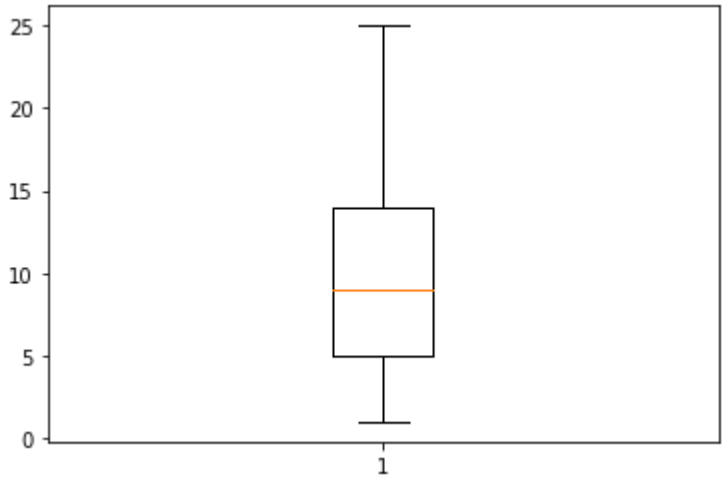
In [236]:
```
y = list(ability_upgrades.ability)
plt.boxplot(y)
plt.show()
```




In [171]:
```
ability_upgrades_feature_2.describe()
```

Out[171]:
```
count    8.939599e+06
mean     9.974747e+00
std      5.963747e+00
min      1.000000e+00
25%      5.000000e+00
50%      9.000000e+00
75%      1.400000e+01
max      2.500000e+01
Name: level, dtype: float64
```

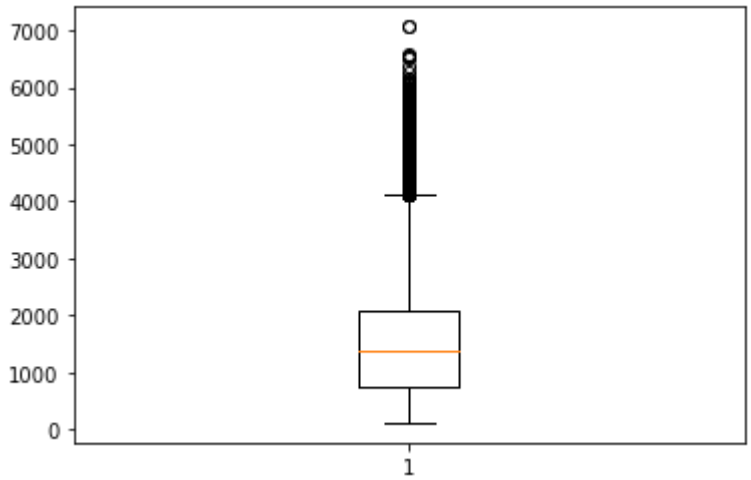
In [237]:
```
y = list(ability_upgrades.level)
plt.boxplot(y)
plt.show()
```




In [172]:
```
ability_upgrades_feature_3.describe()
```

Out[172]:
```
count    8.939599e+06
mean     1.495252e+03
std      8.401830e+02
min      1.240000e+02
25%      7.610000e+02
50%      1.368000e+03
75%      2.103000e+03
max      7.063000e+03
Name: time, dtype: float64
```

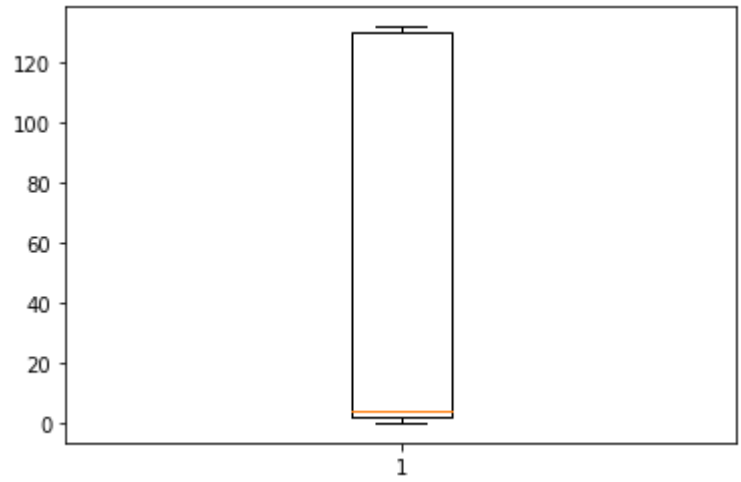
In [238]:
```
y = list(ability_upgrades.time)
plt.boxplot(y)
plt.show()
```



In [173]:
```
ability_upgrades_feature_4.describe()
```

Out[173]:
```
count    8.939599e+06
mean     6.594974e+01
std      6.401578e+01
min      0.000000e+00
25%      2.000000e+00
50%      4.000000e+00
75%      1.300000e+02
max      1.320000e+02
Name: player_slot, dtype: float64
```

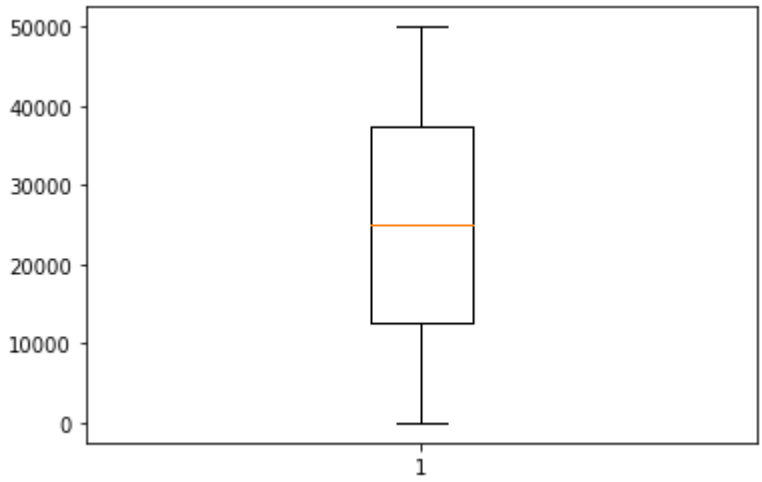
In [239]:
```
y = list(ability_upgrades.player_slot)
plt.boxplot(y)
plt.show()
```




In [103]:
```
ability_upgrades_feature_5.describe()
```

Out[103]:
```
count    8.939599e+06
mean     2.500666e+04
std      1.442761e+04
min      0.000000e+00
25%      1.253000e+04
50%      2.502100e+04
75%      3.749400e+04
max      4.999900e+04
Name: match_id, dtype: float64
```

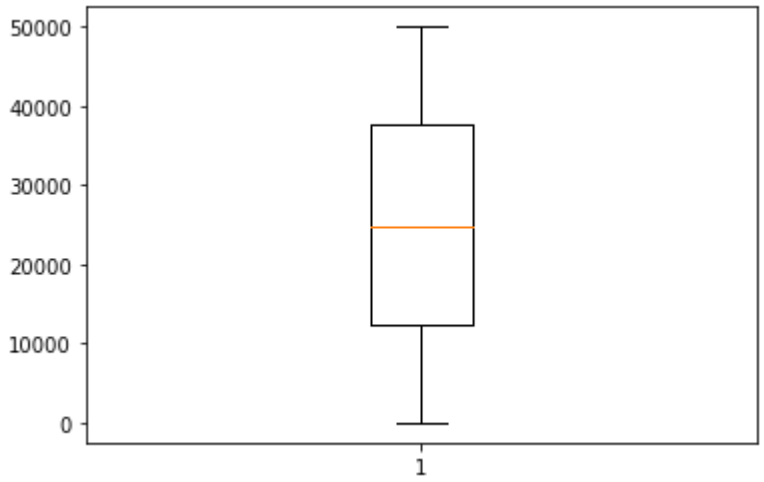
In [240]:
```
y = list(ability_upgrades.match_id)
plt.boxplot(y)
plt.show()
```




In [104]:
```
# chat dataset

chat[:5]

chat_feature_1 = chat["match_id"]
chat_feature_2 = chat["key"]
chat_feature_3 = chat["slot"]
chat_feature_4 = chat["time"]
chat_feature_5 = chat["unit"]

chat_feature_1.describe()
```

Out[104]:
```
count    1.439488e+06
mean     2.495673e+04
std      1.457124e+04
min      0.000000e+00
25%      1.227900e+04
50%      2.483900e+04
75%      3.767800e+04
max      4.999900e+04
Name: match_id, dtype: float64
```

In [241]:
```
y = list(chat.match_id)
plt.boxplot(y)
plt.show()
```



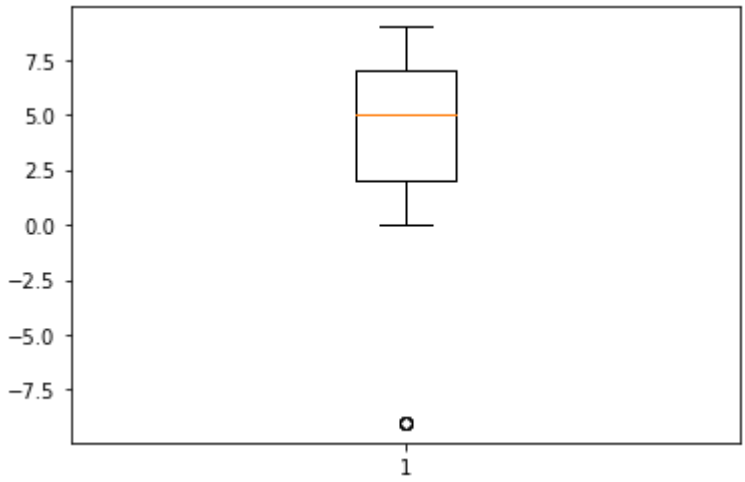

In [105]:
```
chat_feature_2.describe()
```

Out[105]:
```
count     1439474
unique     690814
top            gg
freq        65412
Name: key, dtype: object
```

In [106]:
```
chat_feature_3.describe()
```

Out[106]:
```
count    1.439488e+06
mean     4.493948e+00
std      2.877044e+00
min     -9.000000e+00
25%      2.000000e+00
50%      5.000000e+00
75%      7.000000e+00
max      9.000000e+00
Name: slot, dtype: float64
```

In [244]:
```
y = list(chat.slot)
plt.boxplot(y)
plt.show()
```




In [107]:
```
chat_feature_4.describe()
```

Out[107]:
```
count    1.439488e+06
mean     1.708153e+03
std      9.868280e+02
min     -9.070000e+02
25%      1.020000e+03
50%      1.776000e+03
75%      2.386000e+03
max      1.605700e+04
Name: time, dtype: float64
```

In [ ]:
```
y = list(chat.slot)
plt.boxplot(y)
plt.show()
```

In [108]:
```
chat_feature_5.describe()
```

Out[108]:
```
count       1439446
unique       177991
top       Deez_Nuts
freq           1507
Name: unit, dtype: object
```

In [ ]:
```
y = list(chat.slot)
plt.boxplot(y)
plt.show()
```

In [109]:
```
# cluster_regions dataset

cluster_regions[:5]

cluster_regions_feature_1 = cluster_regions["cluster"]
cluster_regions_feature_2 = cluster_regions["region"]

cluster_regions_feature_1.describe()
```

Out[109]:
```
count     53.000000
mean     174.867925
std       40.967641
min      111.000000
25%      137.000000
50%      181.000000
75%      204.000000
max      261.000000
Name: cluster, dtype: float64
```

In [110]:
```
cluster_regions_feature_2.describe()
```

Out[110]:
```
count            53
unique           20
top       STOCKHOLM
freq              8
Name: region, dtype: object
```

In [111]:
```
# hero_names dataset

hero_names[:5]

hero_names_feature_1 = hero_names["name"]
hero_names_feature_2 = hero_names["hero_id"]
hero_names_feature_3 = hero_names["localized_name"]

hero_names_feature_1.describe()
```

Out[111]:
```
count                        112
unique                       112
top       npc_dota_hero_antimage
freq                           1
Name: name, dtype: object
```

In [112]:
```
hero_names_feature_2.describe()
```

Out[112]:
```
count    112.000000
mean      57.294643
std       32.760842
min        1.000000
25%       29.750000
50%       57.500000
75%       85.250000
max      113.000000
Name: hero_id, dtype: float64
```

In [113]:
```
hero_names_feature_3.describe()
```

Out[113]:
```
count           112
unique          112
top       Anti-Mage
freq              1
Name: localized_name, dtype: object
```

In [114]:
```
# item_ids dataset

item_ids[:5]

item_ids_feature_1 = item_ids["item_id"]
item_ids_feature_2 = item_ids["item_name"]

item_ids_feature_1.describe()
```

Out[114]:
```
count     189.000000
mean      248.772487
std       328.210908
min         1.000000
25%        51.000000
50%       135.000000
75%       229.000000
max      1027.000000
Name: item_id, dtype: float64
```

In [115]:
```
item_ids_feature_2.describe()
```

Out[115]:
```
count       189
unique      189
top       blink
freq          1
Name: item_name, dtype: object
```

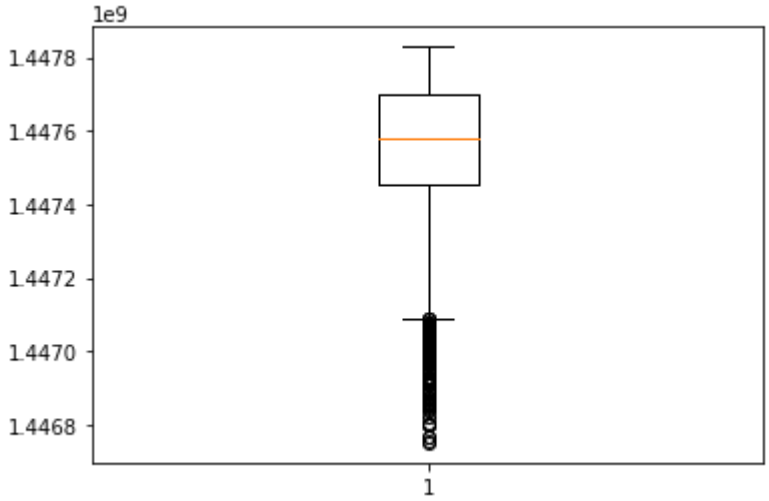
In [116]:
```
# match

match[:5]

match_feature_1 = match["match_id"]
match_feature_2 = match["start_time"]
match_feature_3 = match["duration"]
match_feature_4 = match["tower_status_radiant"]
match_feature_5 = match["tower_status_dire"]
match_feature_6 = match["barracks_status_dire"]
match_feature_7 = match["barracks_status_radiant"]
match_feature_8 = match["first_blood_time"]
match_feature_9 = match["game_mode"]
match_feature_10 = match["radiant_win"]
match_feature_11 = match["negative_votes"]
match_feature_12 = match["positive_votes"]
match_feature_13 = match["cluster"]

match_feature_1.describe()
```

Out[116]:
```
count    50000.000000
mean     24999.500000
std      14433.901067
min          0.000000
25%      12499.750000
50%      24999.500000
75%      37499.250000
max      49999.000000
Name: match_id, dtype: float64
```

In [117]:
```
match_feature_2.describe()
```

Out[117]:
```
count    5.000000e+04
mean     1.447573e+09
std      1.485270e+05
min      1.446750e+09
25%      1.447456e+09
50%      1.447577e+09
75%      1.447700e+09
max      1.447829e+09
Name: start_time, dtype: float64
```

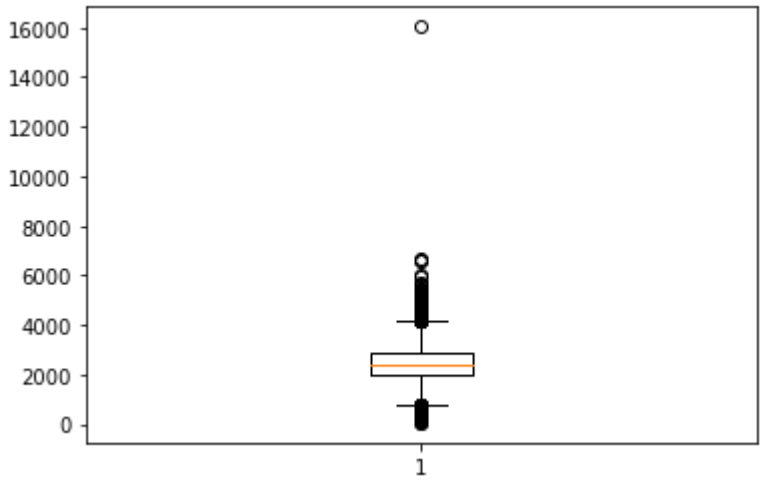
In [247]:
```
y = list(match.start_time)
plt.boxplot(y)
plt.show()
```




In [118]:
```
match_feature_3.describe()
```

Out[118]:
```
count    50000.000000
mean      2476.453500
std        634.631261
min         59.000000
25%       2029.000000
50%       2415.000000
75%       2872.000000
max      16037.000000
Name: duration, dtype: float64
```

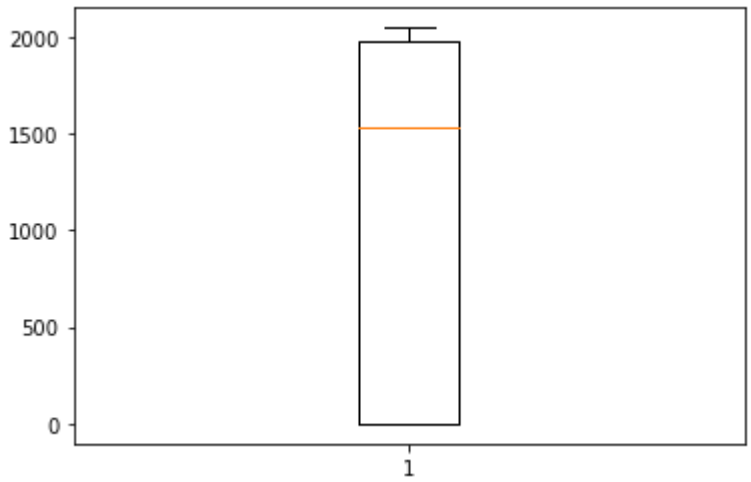
In [248]:
```
y = list(match.duration)
plt.boxplot(y)
plt.show()
```




In [119]:
```
match_feature_4.describe()
```

Out[119]:
```
count    50000.000000
mean      1000.016440
std        948.211846
min          0.000000
25%          0.000000
50%       1536.000000
75%       1974.000000
max       2047.000000
Name: tower_status_radiant, dtype: float64
```

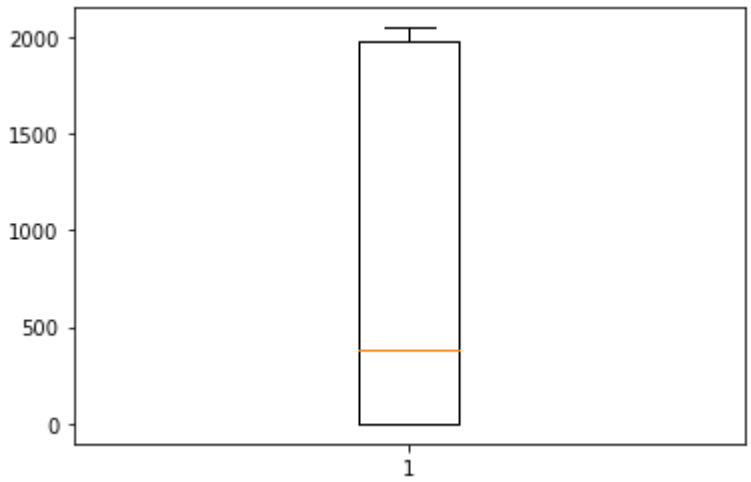
In [249]:
```
y = list(match.tower_status_radiant)
plt.boxplot(y)
plt.show()
```



In [120]:
```
match_feature_5.describe()
```

Out[120]:
```
count    50000.000000
mean       935.250060
std        937.974714
min          0.000000
25%          0.000000
50%        384.000000
75%       1972.000000
max       2047.000000
Name: tower_status_dire, dtype: float64
```

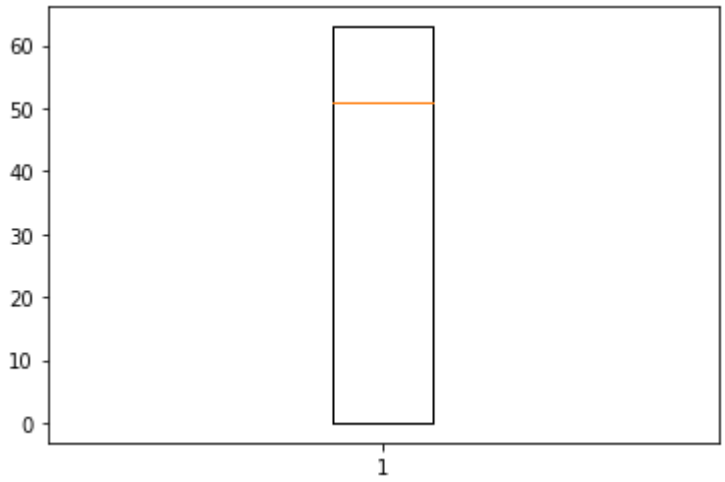
In [250]:
```
y = list(match.tower_status_dire)
plt.boxplot(y)
plt.show()
```




In [121]:
```
match_feature_6.describe()
```

Out[121]:
```
count    50000.000000
mean        34.529460
std         29.209672
min          0.000000
25%          0.000000
50%         51.000000
75%         63.000000
max         63.000000
Name: barracks_status_dire, dtype: float64
```

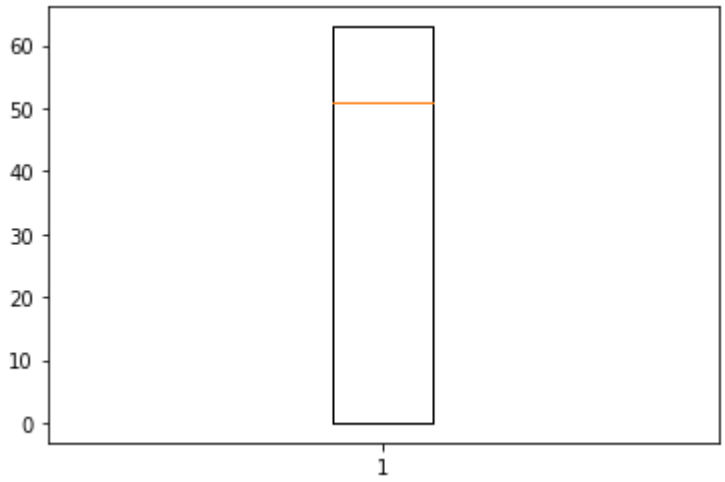
In [251]:
```
y = list(match.barracks_status_dire)
plt.boxplot(y)
plt.show()
```




In [122]:
```
match_feature_7.describe()
```

Out[122]:
```
count    50000.00000
mean        34.77526
std         29.73214
min          0.00000
25%          0.00000
50%         51.00000
75%         63.00000
max         63.00000
Name: barracks_status_radiant, dtype: float64
```

In [252]:
```
y = list(match.barracks_status_radiant)
plt.boxplot(y)
plt.show()
```




In [123]:
```
match_feature_8.describe()
```

Out[123]:
```
count    50000.000000
mean        93.825520
std         92.648332
min          0.000000
25%          9.000000
50%         77.000000
75%        144.000000
max        831.000000
Name: first_blood_time, dtype: float64
```

In [ ]:
```
y = list(match.first_blood_time)
plt.boxplot(y)
plt.show()
```

In [124]:
```
match_feature_9.describe()
```

Out[124]:
```
count    50000.000000
mean        21.468000
std          3.218258
min          2.000000
25%         22.000000
50%         22.000000
75%         22.000000
max         22.000000
Name: game_mode, dtype: float64
```

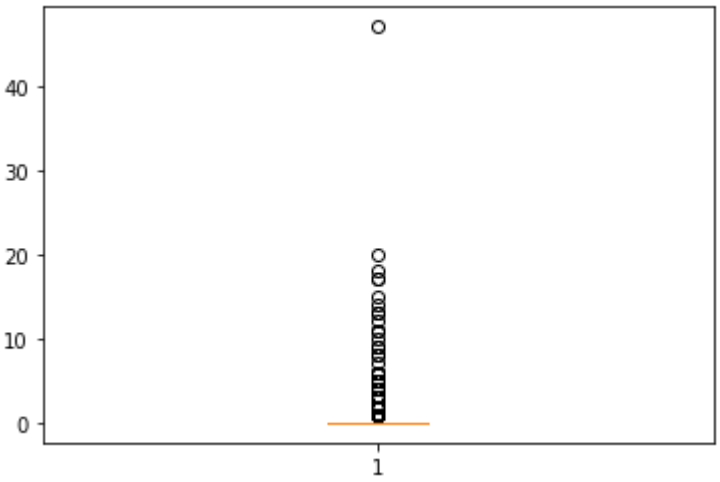
In [125]:
```
match_feature_10.describe()
```

Out[125]:
```
count     50000
unique        2
top        True
freq      25943
Name: radiant_win, dtype: object
```

In [126]:
```
match_feature_11.describe()
```

Out[126]:
```
count    50000.000000
mean         0.015480
std          0.364696
min          0.000000
25%          0.000000
50%          0.000000
75%          0.000000
max         47.000000
Name: negative_votes, dtype: float64
```

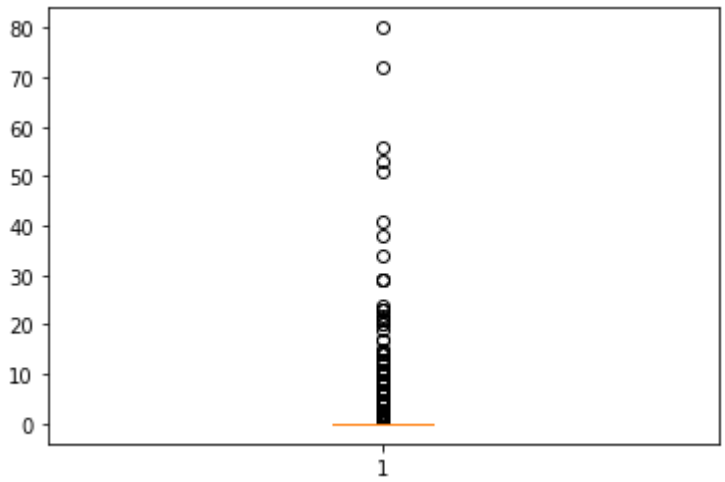
In [255]:
```
y = list(match.negative_votes)
plt.boxplot(y)
plt.show()
```



In [127]:
```
match_feature_12.describe()
```

Out[127]:
```
count    50000.000000
mean         0.036820
std          0.871068
min          0.000000
25%          0.000000
50%          0.000000
75%          0.000000
max         80.000000
Name: positive_votes, dtype: float64
```

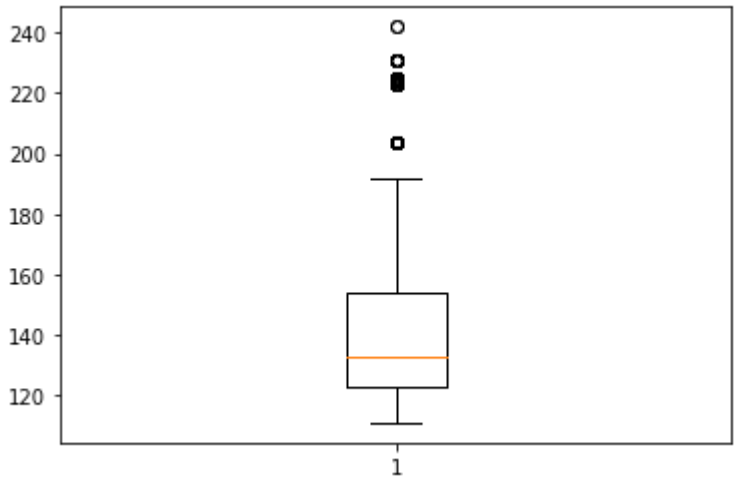
In [256]:
```
y = list(match.positive_votes)
plt.boxplot(y)
plt.show()
```




In [128]:
```
match_feature_13.describe()
```

Out[128]:
```
count    50000.000000
mean       142.304720
std         25.156608
min        111.000000
25%        123.000000
50%        133.000000
75%        154.000000
max        242.000000
Name: cluster, dtype: float64
```

In [257]:
```
y = list(match.cluster)
plt.boxplot(y)
plt.show()
```



In [129]:
```
# objectives dataset

objectives[:5]

objectives_feature_1 = objectives["match_id"]
objectives_feature_2 = objectives["key"]
objectives_feature_3 = objectives["player1"]
objectives_feature_4 = objectives["player2"]
objectives_feature_5 = objectives["slot"]
objectives_feature_6 = objectives["subtype"]
objectives_feature_7 = objectives["team"]
objectives_feature_8 = objectives["time"]
objectives_feature_9 = objectives["value"]

objectives_feature_1.describe()
```

Out[129]:
```
count    1.173396e+06
mean     2.502495e+04
std      1.443535e+04
min      0.000000e+00
25%      1.253500e+04
50%      2.505100e+04
75%      3.752100e+04
max      4.999900e+04
Name: match_id, dtype: float64
```

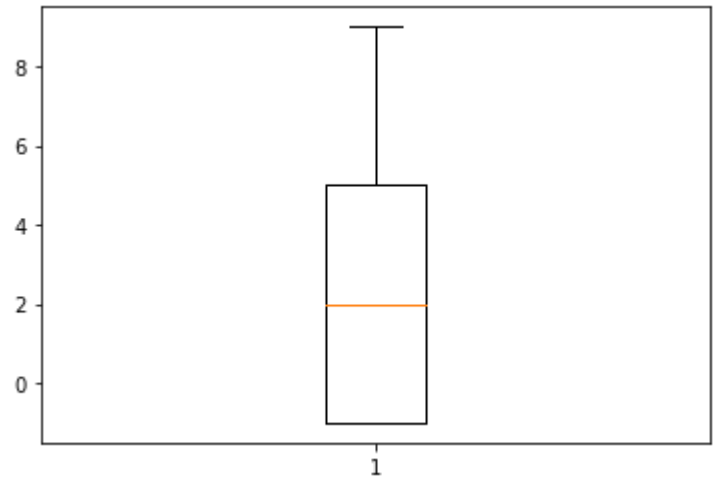
In [130]:
```
objectives_feature_2.describe()
```

Out[130]:
```
count    269576.000000
mean        309.734769
std         552.679820
min           1.000000
25%           4.000000
50%          32.000000
75%         256.000000
max        2048.000000
Name: key, dtype: float64
```

In [131]:
```
objectives_feature_3.describe()
```

Out[131]:
```
count    1.173396e+06
mean     2.236327e+00
std      3.376540e+00
min     -1.000000e+00
25%     -1.000000e+00
50%      2.000000e+00
75%      5.000000e+00
max      9.000000e+00
Name: player1, dtype: float64
```

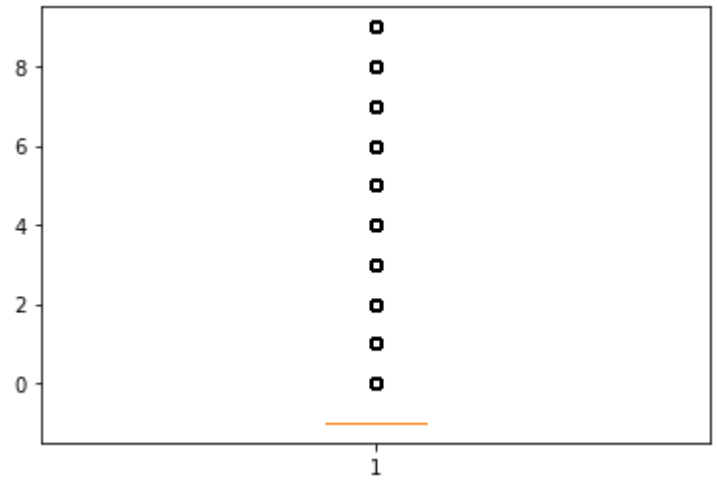
In [258]:
```
y = list(objectives.player1)
plt.boxplot(y)
plt.show()
```




In [132]:
```
objectives_feature_4.describe()
```

Out[132]:
```
count    1.173396e+06
mean    -7.733783e-01
std      1.237207e+00
min     -1.000000e+00
25%     -1.000000e+00
50%     -1.000000e+00
75%     -1.000000e+00
max      9.000000e+00
Name: player2, dtype: float64
```

In [259]:
```
y = list(objectives.player2)
plt.boxplot(y)
plt.show()
```




In [133]:
```
objectives_feature_5.describe()
```

Out[133]:
```
count    826853.000000
mean          3.257962
std           3.418533
min          -1.000000
25%           0.000000
50%           3.000000
75%           6.000000
max           9.000000
Name: slot, dtype: float64
```

In [134]:
```
objectives_feature_6.describe()
```

Out[134]:
```
count                    1173396
unique                         7
top       CHAT_MESSAGE_TOWER_KILL
freq                      663032
Name: subtype, dtype: object
```

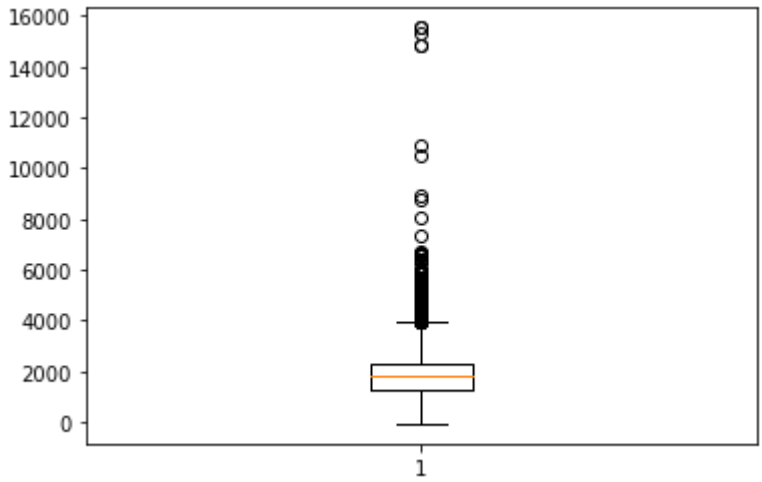
In [135]:
```
objectives_feature_7.describe()
```

Out[135]:
```
count    778755.000000
mean          6.642750
std          18.291118
min           2.000000
25%           2.000000
50%           3.000000
75%           3.000000
max         140.000000
Name: team, dtype: float64
```

In [136]:
```
objectives_feature_8.describe()
```

Out[136]:
```
count    1.173396e+06
mean     1.799777e+03
std      7.932244e+02
min     -5.500000e+01
25%      1.255000e+03
50%      1.805000e+03
75%      2.332000e+03
max      1.552900e+04
Name: time, dtype: float64
```

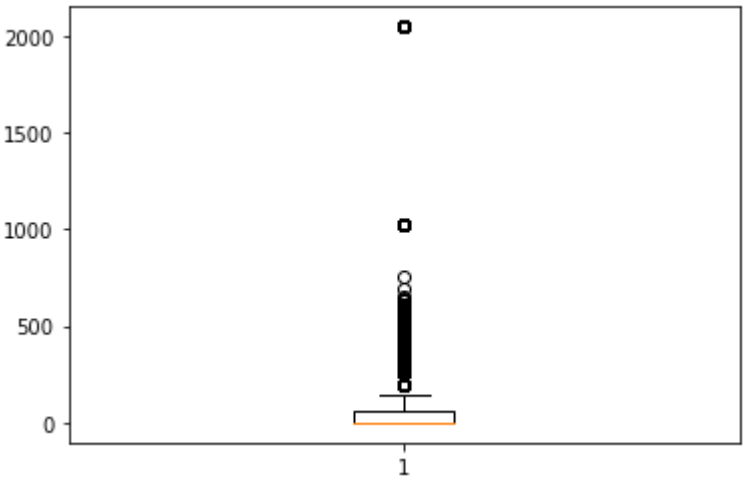
In [263]:
```
y = list(objectives.time)
plt.boxplot(y)
plt.show()
```




In [137]:
```
objectives_feature_9.describe()
```

Out[137]:
```
count    1.173396e+06
mean     1.034998e+02
std      2.999297e+02
min      0.000000e+00
25%      2.000000e+00
50%      3.000000e+00
75%      6.400000e+01
max      2.048000e+03
Name: value, dtype: float64
```

In [264]:
```
y = list(objectives.value)
plt.boxplot(y)
plt.show()
```




In [138]:
```
# patch_dates dataset

patch_dates

patch_dates_feature_1 = patch_dates["patch_date"]
patch_dates_feature_2 = patch_dates["name"]

patch_dates_feature_1.describe()
```

Out[138]:
```
count                         19
unique                        19
top       2010-12-24T00:00:00Z
freq                           1
Name: patch_date, dtype: object
```

In [139]:
```
patch_dates_feature_2.describe()
```

Out[139]:
```
count    19.000000
mean      6.790000
std       0.056273
min       6.700000
25%       6.745000
50%       6.790000
75%       6.835000
max       6.880000
Name: name, dtype: float64
```

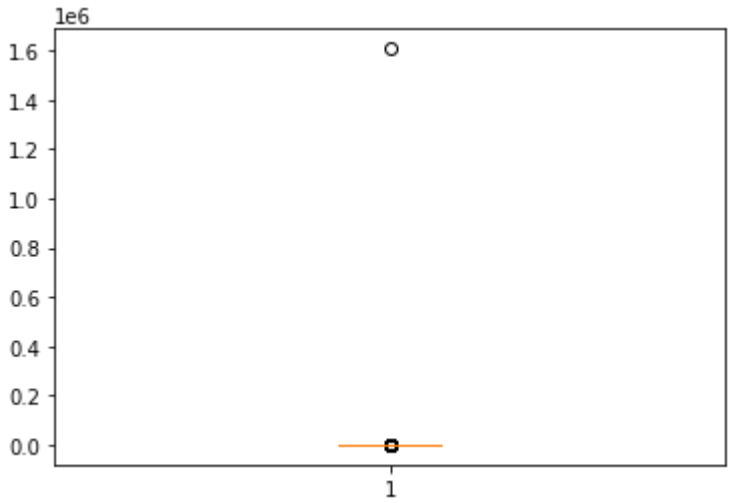
In [140]:
```
# player_ratings dataset

player_ratings

player_ratings_feature_1 = player_ratings["account_id"]
player_ratings_feature_2 = player_ratings["total_wins"]
player_ratings_feature_3 = player_ratings["total_matches"]
player_ratings_feature_4 = player_ratings["trueskill_mu"]
player_ratings_feature_5 = player_ratings["trueskill_sigma"]

player_ratings_feature_1.describe()
```

Out[140]:
```
count    8.342260e+05
mean    -9.225868e+07
std      8.103222e+07
min     -2.991940e+08
25%     -1.499249e+08
50%     -9.585022e+07
75%      4.883475e+04
max      3.305130e+05
Name: account_id, dtype: float64
```

In [141]:
```
player_ratings_feature_2.describe()
```

Out[141]:
```
count    8.342260e+05
mean     5.479852e+00
std      1.760984e+03
min      0.000000e+00
25%      0.000000e+00
50%      1.000000e+00
75%      3.000000e+00
max      1.608398e+06
Name: total_wins, dtype: float64
```

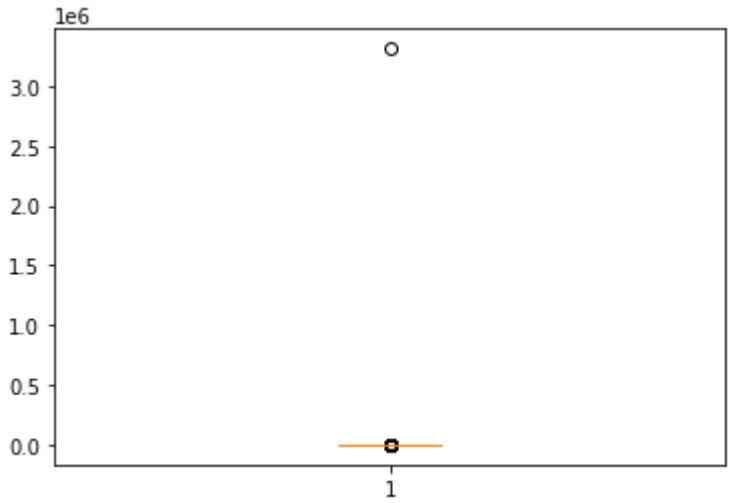
In [265]:
```
y = list(player_ratings.total_wins)
plt.boxplot(y)
plt.show()
```




In [142]:
```
player_ratings_feature_3.describe()
```

Out[142]:
```
count    8.342260e+05
mean     1.095979e+01
std      3.629559e+03
min      1.000000e+00
25%      1.000000e+00
50%      2.000000e+00
75%      6.000000e+00
max      3.315071e+06
Name: total_matches, dtype: float64
```

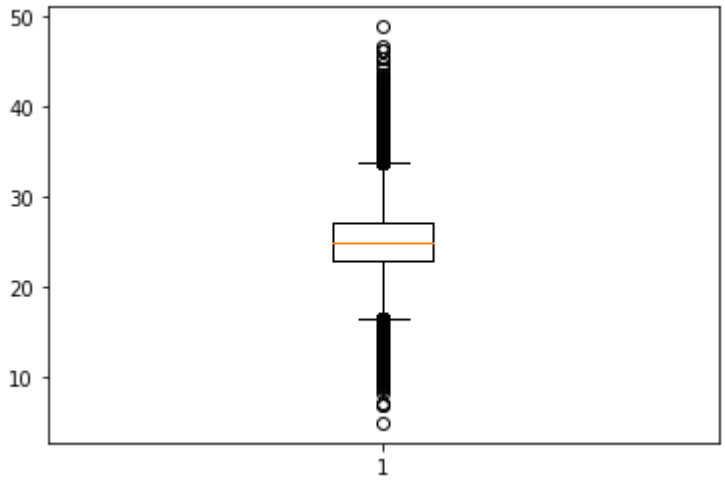
In [266]:
```
y = list(player_ratings.total_matches)
plt.boxplot(y)
plt.show()
```




In [143]:
```
player_ratings_feature_4.describe()
```

Out[143]:
```
count    834226.000000
mean         25.112577
std           3.231603
min           4.993478
25%          22.906655
50%          25.018193
75%          27.240350
max          48.825892
Name: trueskill_mu, dtype: float64
```

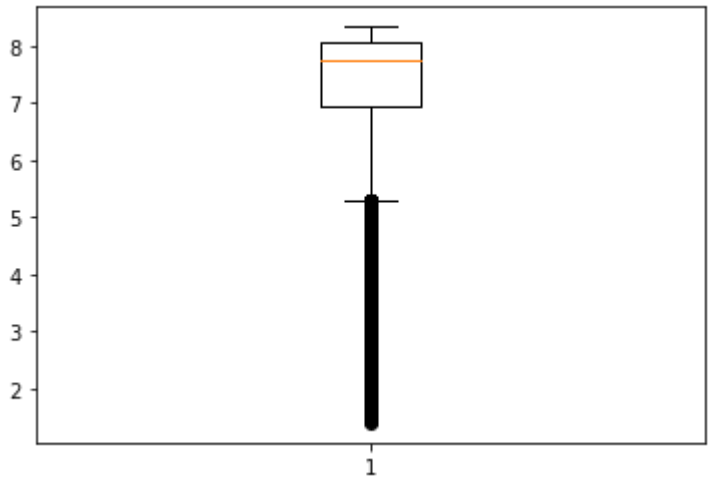
In [267]:
```
y = list(player_ratings.trueskill_mu)
plt.boxplot(y)
plt.show()
```




In [144]:
```
player_ratings_feature_5.describe()
```

Out[144]:
```
count    834226.000000
mean          7.270275
std           1.128826
min           1.404098
25%           6.957458
50%           7.732504
75%           8.058739
max           8.333689
Name: trueskill_sigma, dtype: float64
```

In [268]:
```
y = list(player_ratings.trueskill_sigma)
plt.boxplot(y)
plt.show()
```



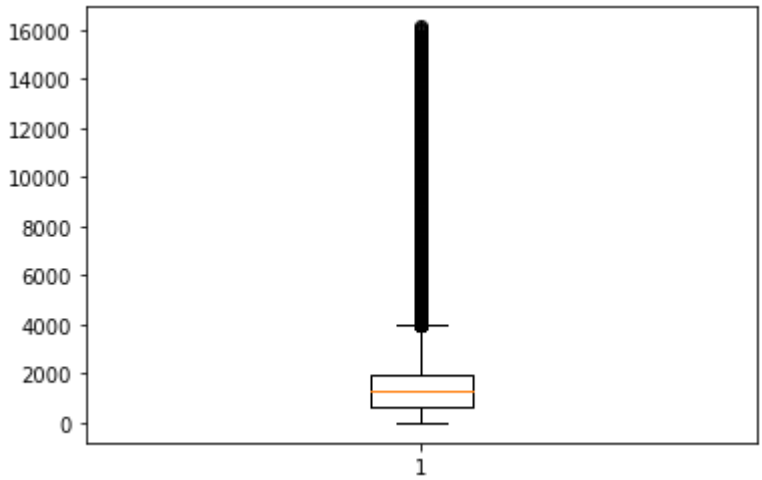

In [145]:
```
# player_time

player_time

player_time_feature_1 = player_time["match_id"]
player_time_feature_2 = player_time["times"]
player_time_feature_3 = player_time["gold_t_0"]
player_time_feature_4 = player_time["lh_t_0"]
player_time_feature_5 = player_time["xp_t_0"]
player_time_feature_6 = player_time["gold_t_1"]
player_time_feature_7 = player_time["lh_t_1"]
player_time_feature_8 = player_time["xp_t_1"]
player_time_feature_9 = player_time["gold_t_2"]
player_time_feature_10 = player_time["lh_t_2"]
player_time_feature_11 = player_time["xp_t_129"]
player_time_feature_12 = player_time["gold_t_130"]
player_time_feature_13 = player_time["lh_t_130"]
player_time_feature_14 = player_time["xp_t_130"]
player_time_feature_15 = player_time["gold_t_131"]
player_time_feature_16 = player_time["lh_t_131"]
player_time_feature_17 = player_time["gold_t_132"]
player_time_feature_18 = player_time["lh_t_132"]
player_time_feature_19 = player_time["xp_t_132"]

player_time_feature_1.describe()
```

Out[145]:
```
count    2.209778e+06
mean     2.501692e+04
std      1.443619e+04
min      0.000000e+00
25%      1.253500e+04
50%      2.503200e+04
75%      3.752300e+04
max      4.999900e+04
Name: match_id, dtype: float64
```

In [146]:
```
player_time_feature_2.describe()
```

Out[146]:
```
count    2.209778e+06
mean     1.371875e+03
std      8.965257e+02
min      0.000000e+00
25%      6.600000e+02
50%      1.320000e+03
75%      1.980000e+03
max      1.614000e+04
Name: times, dtype: float64
```

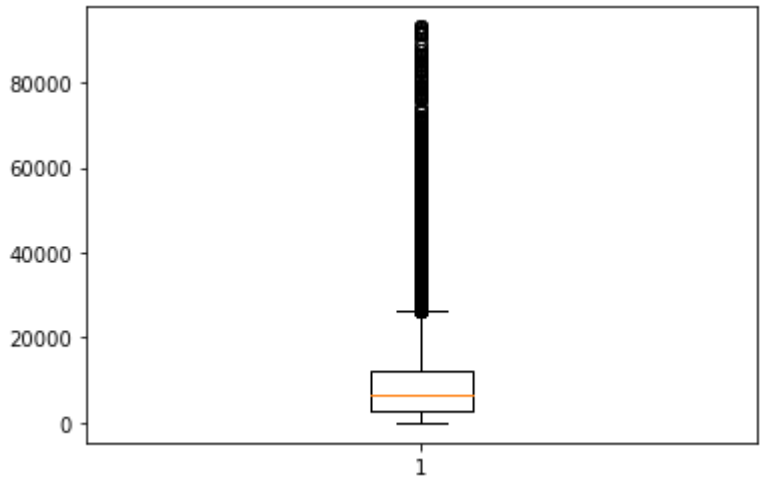
In [277]:
```
y = list(player_time.times)
plt.boxplot(y)
plt.show()
```




In [270]:
```
player_time_feature_3.describe()
```

Out[270]:
```
count    2.209778e+06
mean     8.214262e+03
std      7.023217e+03
min      0.000000e+00
25%      2.589000e+03
50%      6.631000e+03
75%      1.206300e+04
max      9.317100e+04
Name: gold_t_0, dtype: float64
```

In [278]:
```
y = list(player_time.gold_t_0)
plt.boxplot(y)
plt.show()
```




In [279]:
```
player_time_feature_4.describe()
```

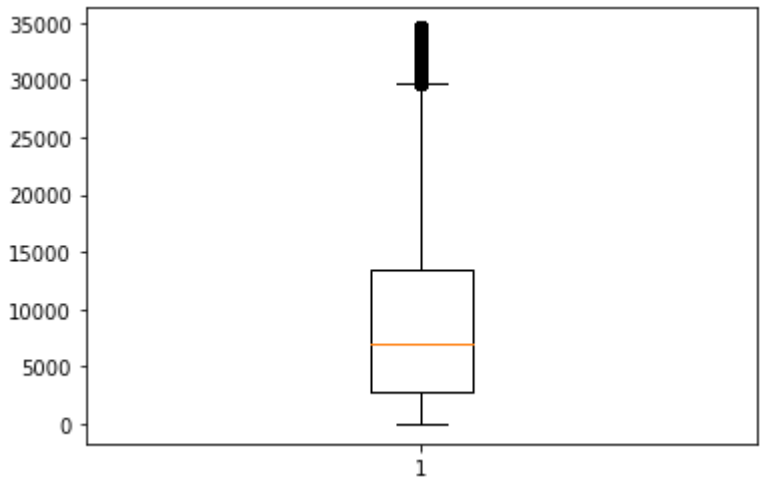
Out[279]:
```
count    2.209778e+06
mean     6.858699e+01
std      7.571156e+01
min      0.000000e+00
25%      1.500000e+01
50%      4.400000e+01
75%      9.700000e+01
max      1.183000e+03
Name: lh_t_0, dtype: float64
```

In [ ]:
```
y = list(player_time.lh_t_0)
plt.boxplot(y)
plt.show()
```

In [149]:
```
player_time_feature_5.describe()
```

Out[149]:
```
count    2.209778e+06
mean     9.169268e+03
std      8.035813e+03
min      0.000000e+00
25%      2.726000e+03
50%      7.034000e+03
75%      1.351200e+04
max      3.457400e+04
Name: xp_t_0, dtype: float64
```

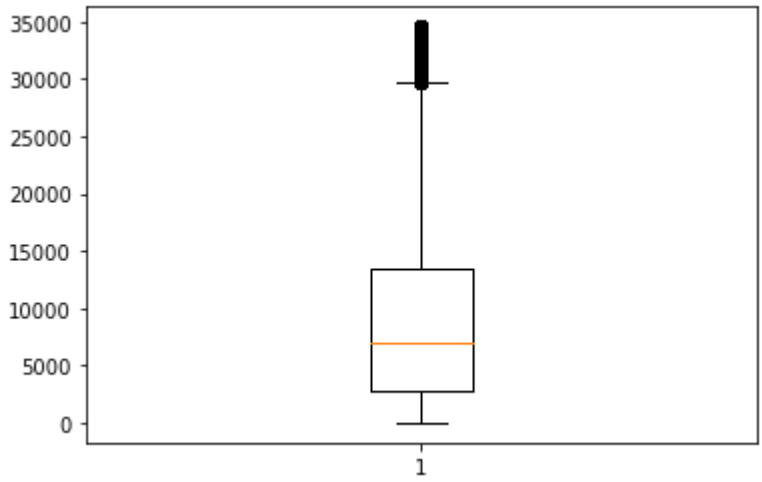
In [280]:
```
y = list(player_time.xp_t_0)
plt.boxplot(y)
plt.show()
```




In [150]:
```
player_time_feature_6.describe()
```

Out[150]:
```
count    2.209778e+06
mean     8.073114e+03
std      6.968762e+03
min      0.000000e+00
25%      2.536000e+03
50%      6.514000e+03
75%      1.183700e+04
max      1.727280e+05
Name: gold_t_1, dtype: float64
```

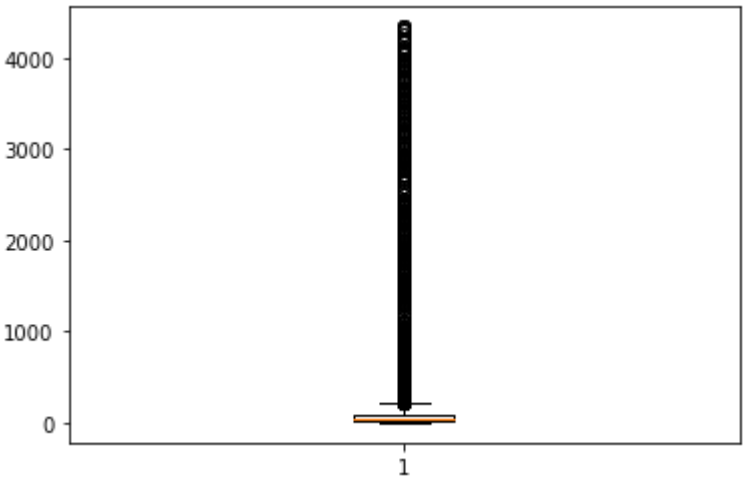
In [274]:
```
y = list(player_time.gold_t_1)
plt.boxplot(y)
plt.show()
```




In [151]:
```
player_time_feature_7.describe()
```

Out[151]:
```
count    2.209778e+06
mean     6.630451e+01
std      7.815977e+01
min      0.000000e+00
25%      1.400000e+01
50%      4.200000e+01
75%      9.300000e+01
max      4.341000e+03
Name: lh_t_1, dtype: float64
```

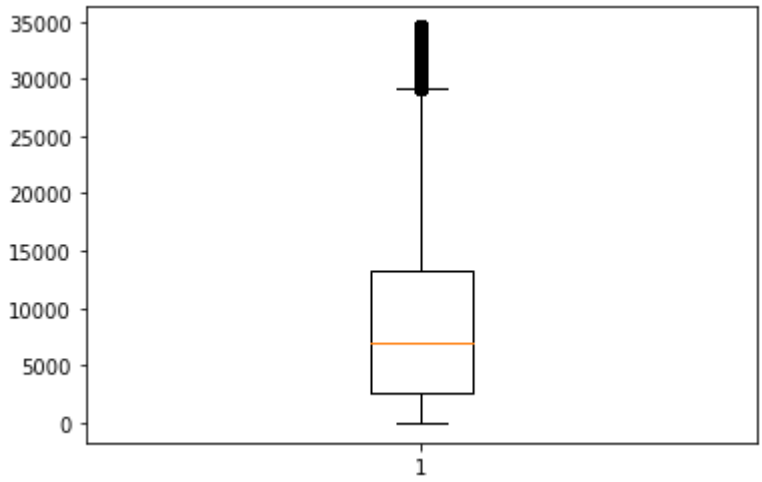
In [281]:
```
y = list(player_time.lh_t_1)
plt.boxplot(y)
plt.show()
```




In [152]:
```
player_time_feature_8.describe()
```

Out[152]:
```
count    2.209778e+06
mean     9.003470e+03
std      7.923962e+03
min      0.000000e+00
25%      2.668000e+03
50%      6.899000e+03
75%      1.323800e+04
max      3.455100e+04
Name: xp_t_1, dtype: float64
```

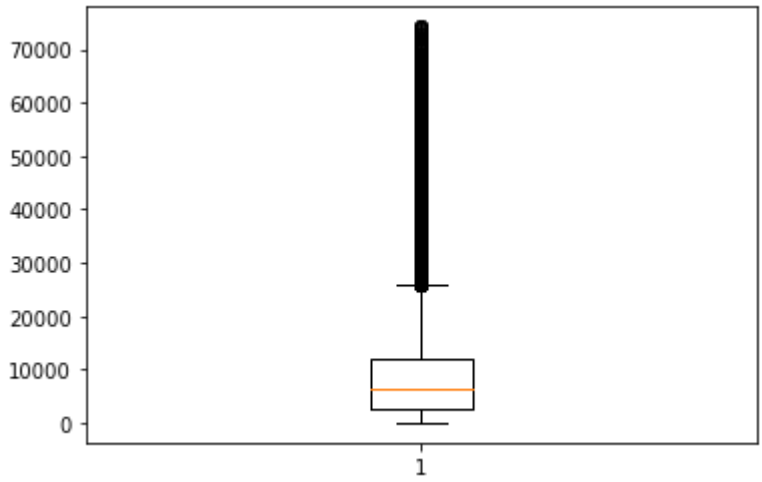
In [282]:
```
y = list(player_time.xp_t_1)
plt.boxplot(y)
plt.show()
```




In [153]:
```
player_time_feature_9.describe()
```

Out[153]:
```
count    2.209778e+06
mean     8.082828e+03
std      6.924552e+03
min      0.000000e+00
25%      2.536000e+03
50%      6.514000e+03
75%      1.185700e+04
max      7.424700e+04
Name: gold_t_2, dtype: float64
```

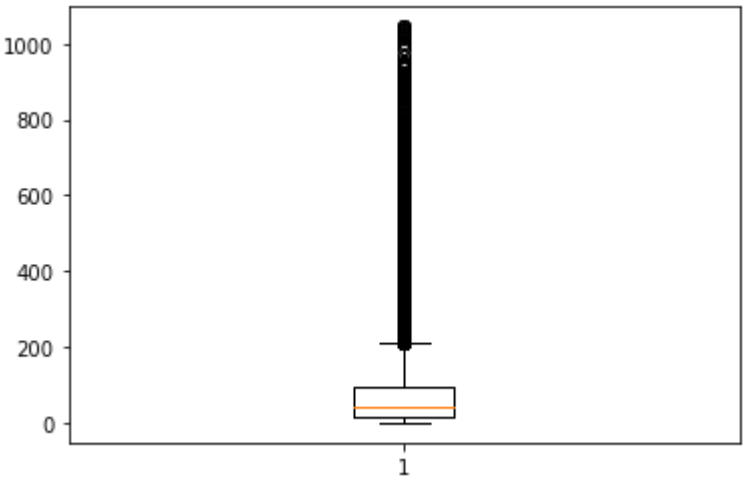
In [283]:
```
y = list(player_time.gold_t_2)
plt.boxplot(y)
plt.show()
```




In [154]:
```
player_time_feature_10.describe()
```

Out[154]:
```
count    2.209778e+06
mean     6.627810e+01
std      7.398861e+01
min      0.000000e+00
25%      1.400000e+01
50%      4.200000e+01
75%      9.300000e+01
max      1.045000e+03
Name: lh_t_2, dtype: float64
```

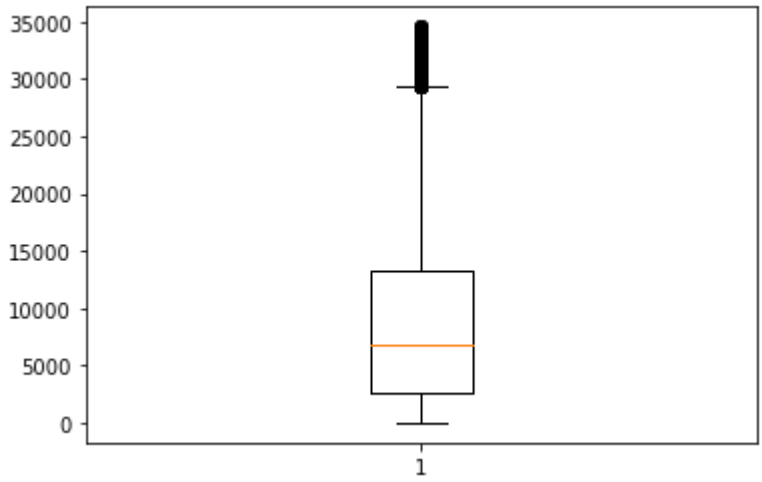
In [284]:
```
y = list(player_time.lh_t_2)
plt.boxplot(y)
plt.show()
```




In [155]:
```
player_time_feature_11.describe()
```

Out[155]:
```
count    2.209778e+06
mean     9.087373e+03
std      8.051436e+03
min      0.000000e+00
25%      2.662000e+03
50%      6.885000e+03
75%      1.336000e+04
max      3.457100e+04
Name: xp_t_129, dtype: float64
```

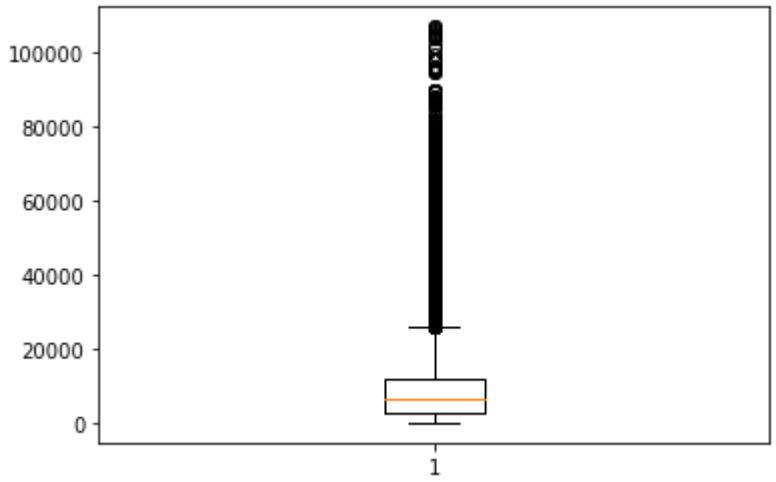
In [285]:
```
y = list(player_time.xp_t_129)
plt.boxplot(y)
plt.show()
```




In [156]:
```
player_time_feature_12.describe()
```

Out[156]:
```
count    2.209778e+06
mean     8.103465e+03
std      7.035336e+03
min      0.000000e+00
25%      2.530000e+03
50%      6.460000e+03
75%      1.183800e+04
max      1.070560e+05
Name: gold_t_130, dtype: float64
```

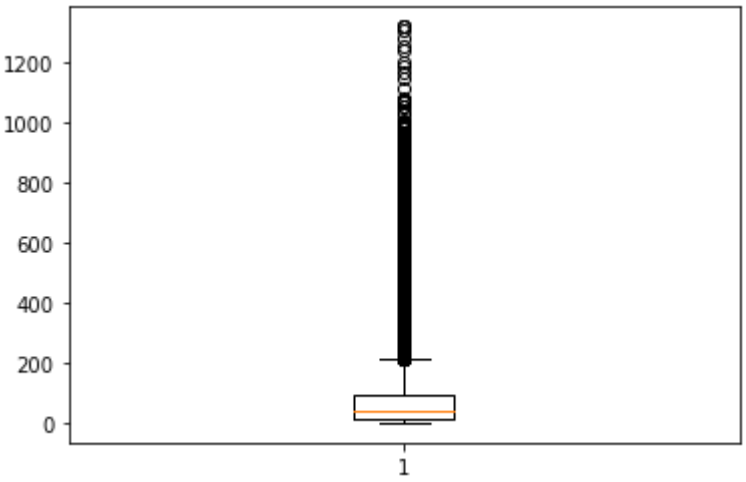
In [286]:
```
y = list(player_time.gold_t_130)
plt.boxplot(y)
plt.show()
```




In [157]:
```
player_time_feature_13.describe()
```

Out[157]:
```
count    2.209778e+06
mean     6.671832e+01
std      7.539959e+01
min      0.000000e+00
25%      1.400000e+01
50%      4.200000e+01
75%      9.300000e+01
max      1.319000e+03
Name: lh_t_130, dtype: float64
```

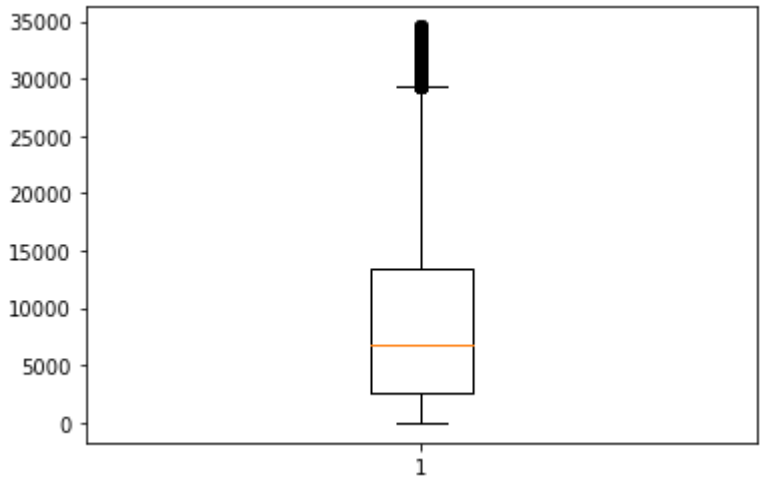
In [287]:
```
y = list(player_time.lh_t_130)
plt.boxplot(y)
plt.show()
```




In [158]:
```
player_time_feature_14.describe()
```

Out[158]:
```
count    2.209778e+06
mean     9.086552e+03
std      8.050322e+03
min      0.000000e+00
25%      2.671000e+03
50%      6.881000e+03
75%      1.334900e+04
max      3.452200e+04
Name: xp_t_130, dtype: float64
```

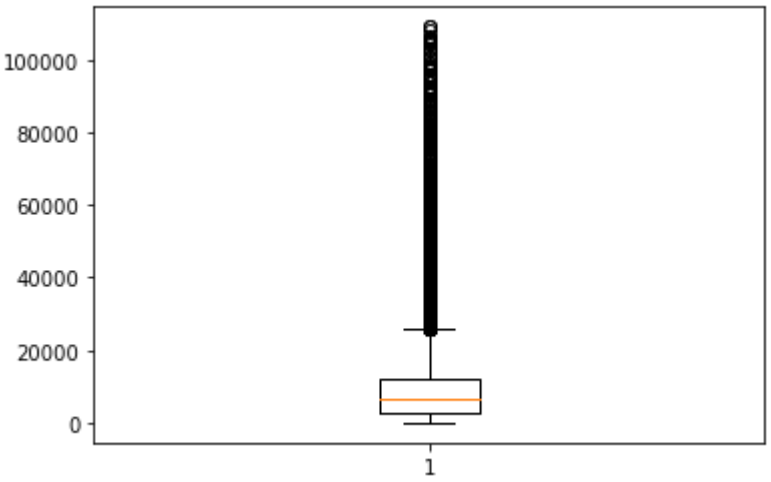
In [288]:
```
y = list(player_time.xp_t_130)
plt.boxplot(y)
plt.show()
```




In [159]:
```
player_time_feature_15.describe()
```

Out[159]:
```
count    2.209778e+06
mean     8.116964e+03
std      7.058066e+03
min      0.000000e+00
25%      2.532000e+03
50%      6.461000e+03
75%      1.186300e+04
max      1.091030e+05
Name: gold_t_131, dtype: float64
```

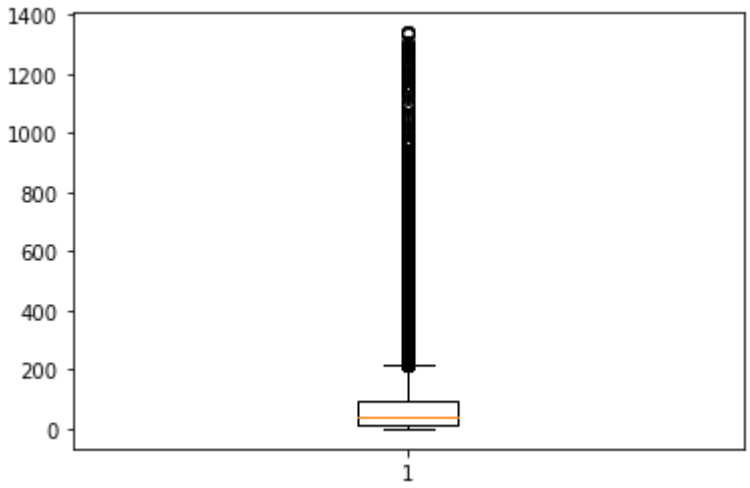
In [289]:
```
y = list(player_time.gold_t_131)
plt.boxplot(y)
plt.show()
```




In [160]:
```
player_time_feature_16.describe()
```

Out[160]:
```
count    2.209778e+06
mean     6.685427e+01
std      7.575212e+01
min      0.000000e+00
25%      1.400000e+01
50%      4.200000e+01
75%      9.400000e+01
max      1.340000e+03
Name: lh_t_131, dtype: float64
```

In [290]:
```
y = list(player_time.lh_t_131)
plt.boxplot(y)
plt.show()
```




In [161]:
```
player_time_feature_17.describe()
```

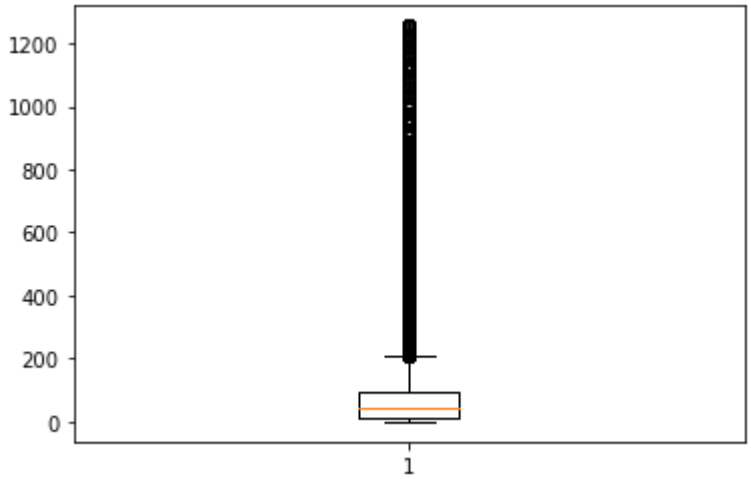
Out[161]:
```
count    2.209778e+06
mean     8.037616e+03
std      6.978853e+03
min      0.000000e+00
25%      2.504000e+03
50%      6.400000e+03
75%      1.174300e+04
max      1.138630e+05
Name: gold_t_132, dtype: float64
```

In [ ]:
```
y = list(player_time.gold_t_132)
plt.boxplot(y)
plt.show()
```

In [162]:
```
player_time_feature_18.describe()
```

Out[162]:
```
count    2.209778e+06
mean     6.585997e+01
std      7.478857e+01
min      0.000000e+00
25%      1.400000e+01
50%      4.100000e+01
75%      9.200000e+01
max      1.256000e+03
Name: lh_t_132, dtype: float64
```

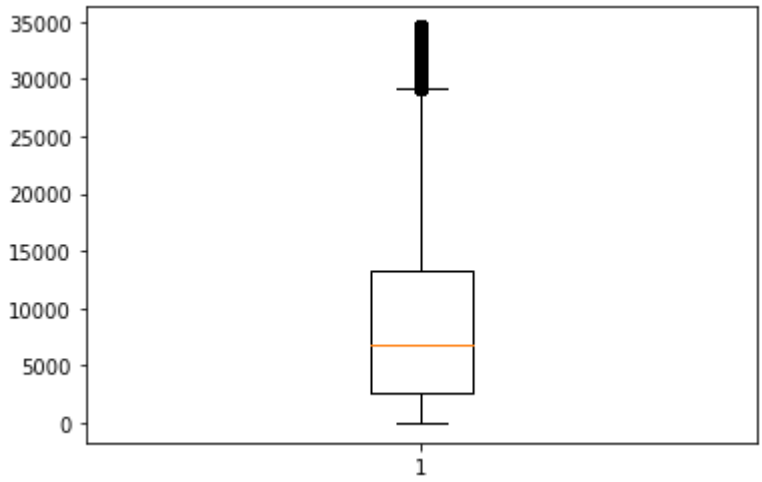
In [291]:
```
y = list(player_time.lh_t_132)
plt.boxplot(y)
plt.show()
```




In [163]:
```
player_time_feature_19.describe()
```

Out[163]:
```
count    2.209778e+06
mean     9.035641e+03
std      8.027818e+03
min      0.000000e+00
25%      2.652000e+03
50%      6.831000e+03
75%      1.325200e+04
max      3.457400e+04
Name: xp_t_132, dtype: float64
```

In [292]:
```
y = list(player_time.xp_t_132)
plt.boxplot(y)
plt.show()
```



In [178]:
```
players

players_feature_1 = players["match_id"]
players_feature_2 = players["account_id"]
players_feature_3 = players["hero_id"]
players_feature_4 = players["player_slot"]
players_feature_5 = players["gold"]
players_feature_6 = players["gold_spent"]
players_feature_7 = players["gold_per_min"]
players_feature_8 = players["kills"]
players_feature_9 = players["deaths"]
players_feature_10 = players["unit_order_glyph"]
players_feature_11 = players["unit_order_eject_item_from_stash"]
players_feature_12 = players["unit_order_cast_rune"]
players_feature_13 = players["unit_order_ping_ability"]
players_feature_14 = players["unit_order_move_to_direction"]
players_feature_15 = players["unit_order_patrol"]
players_feature_16 = players["unit_order_vector_target_position"]
players_feature_17 = players["unit_order_radar"]
players_feature_18 = players["unit_order_set_item_combine_lock"]
players_feature_19 = players["unit_order_continue"]

player_time_feature_1.describe()
```

Out[178]:
```
count    2.209778e+06
mean     2.501692e+04
std      1.443619e+04
min      0.000000e+00
25%      1.253500e+04
50%      2.503200e+04
75%      3.752300e+04
max      4.999900e+04
Name: match_id, dtype: float64
```

In [184]:
```
players_feature_2.describe()
```

Out[184]:
```
count    500000.000000
mean      39589.991014
std       46761.698967
min           0.000000
25%           0.000000
50%       19212.500000
75%       70638.000000
max      158360.000000
Name: account_id, dtype: float64
```

In [185]:
```
players_feature_3.describe()
```

Out[185]:
```
count    500000.000000
mean         50.551456
std          32.809123
min           0.000000
25%          21.000000
50%          47.000000
75%          75.000000
max         112.000000
Name: hero_id, dtype: float64
```

In [186]:
```
players_feature_4.describe()
```

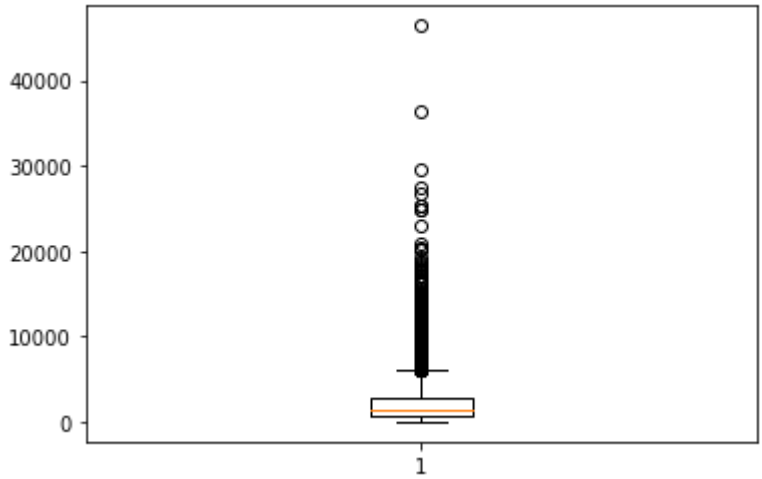
Out[186]:
```
count    500000.000000
mean         66.000000
std          64.015687
min           0.000000
25%           2.000000
50%          66.000000
75%         130.000000
max         132.000000
Name: player_slot, dtype: float64
```

In [293]:
```
y = list(players.player_slot)
plt.boxplot(y)
plt.show()
```

In [187]:
```
players_feature_5.describe()
```

Out[187]:
```
count    500000.000000
mean       1888.516638
std        1742.949262
min           0.000000
25%         586.000000
50%        1350.000000
75%        2742.000000
max       46424.000000
Name: gold, dtype: float64
```

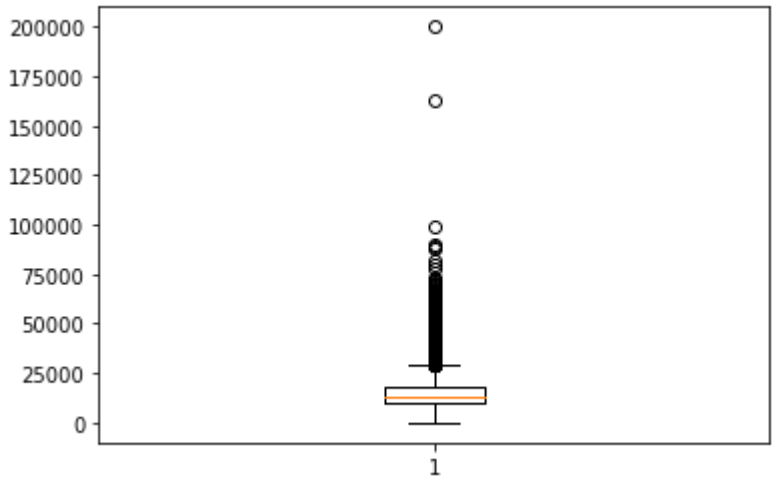
In [294]:
```
y = list(players.gold)
plt.boxplot(y)
plt.show()
```




In [188]:
```
players_feature_6.describe()
```

Out[188]:
```
count    500000.000000
mean      14110.043850
std        6401.868898
min           0.000000
25%        9590.000000
50%       13110.000000
75%       17635.000000
max      200000.000000
Name: gold_spent, dtype: float64
```

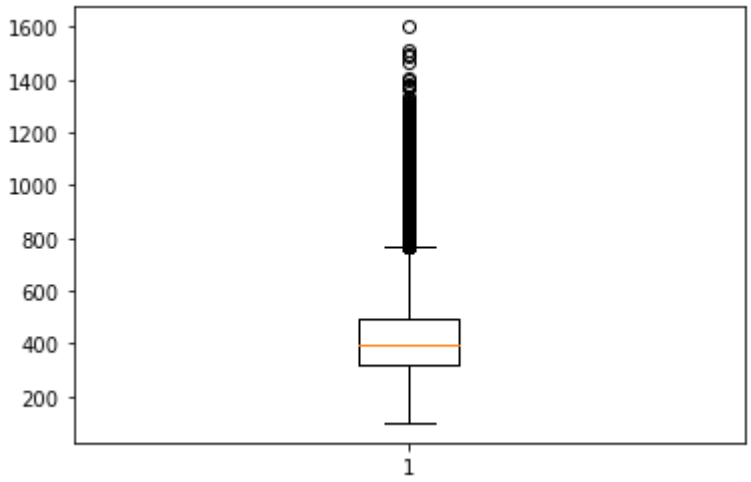
In [295]:
```
y = list(players.gold_spent)
plt.boxplot(y)
plt.show()
```




In [189]:
```
players_feature_7.describe()
```

Out[189]:
```
count    500000.000000
mean        415.097178
std         138.210124
min         100.000000
25%         317.000000
50%         395.000000
75%         496.000000
max        1601.000000
Name: gold_per_min, dtype: float64
```

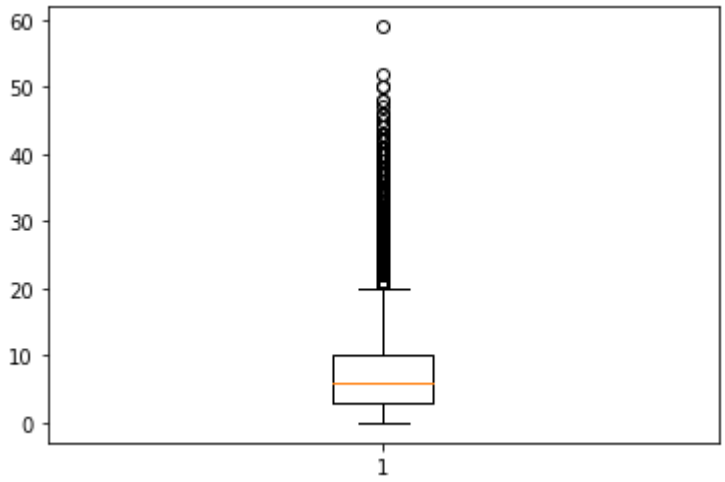
In [296]:
```
y = list(players.gold_per_min)
plt.boxplot(y)
plt.show()
```




In [190]:
```
players_feature_8.describe()
```

Out[190]:
```
count    500000.000000
mean          7.404500
std           5.437802
min           0.000000
25%           3.000000
50%           6.000000
75%          10.000000
max          59.000000
Name: kills, dtype: float64
```

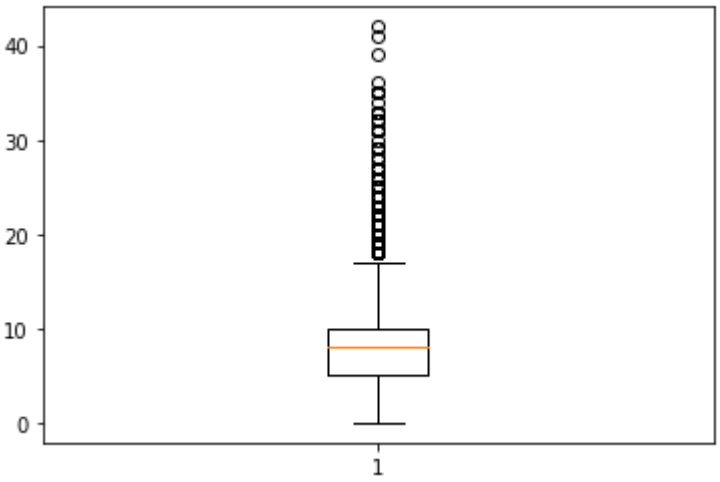
In [297]:
```
y = list(players.kills)
plt.boxplot(y)
plt.show()
```




In [191]:
```
players_feature_9.describe()
```

Out[191]:
```
count    500000.000000
mean          7.680752
std           3.808092
min           0.000000
25%           5.000000
50%           8.000000
75%          10.000000
max          42.000000
Name: deaths, dtype: float64
```

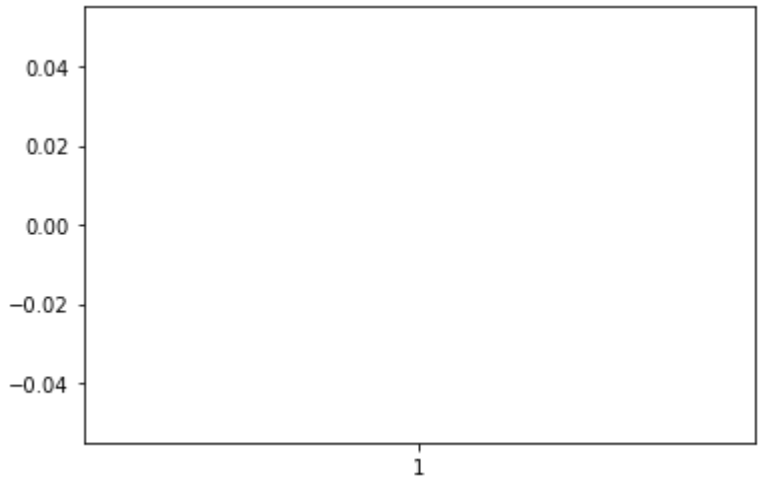
In [298]:
```
y = list(players.deaths)
plt.boxplot(y)
plt.show()
```




In [299]:
```
players_feature_11.describe()
```

Out[299]:
```
count    31264.000000
mean         1.269991
std          0.798112
min          1.000000
25%          1.000000
50%          1.000000
75%          1.000000
max         25.000000
Name: unit_order_eject_item_from_stash, dtype: float64
```

In [301]:
```
y = list(players.unit_order_glyph)
plt.boxplot(y)
plt.show()
```




In [193]:
```
players_feature_11.describe()
```

Out[193]:
```
count    31264.000000
mean         1.269991
std          0.798112
min          1.000000
25%          1.000000
50%          1.000000
75%          1.000000
max         25.000000
Name: unit_order_eject_item_from_stash, dtype: float64
```

In [194]:
```
players_feature_12.describe()
```

Out[194]:
```
count    9.000000
mean     1.222222
std      0.666667
min      1.000000
25%      1.000000
50%      1.000000
75%      1.000000
max      3.000000
Name: unit_order_cast_rune, dtype: float64
```

In [195]:
```
players_feature_13.describe()
```

Out[195]:
```
count    339148.000000
mean          6.197498
std           7.353832
min           1.000000
25%           2.000000
50%           4.000000
75%           8.000000
max         308.000000
Name: unit_order_ping_ability, dtype: float64
```

In [196]:
```
players_feature_14.describe()
```

Out[196]:
```
count    3551.000000
mean       43.181076
std       106.109131
min         1.000000
25%         3.000000
50%         9.000000
75%        36.000000
max      2349.000000
Name: unit_order_move_to_direction, dtype: float64
```

In [197]:
```
players_feature_15.describe()
```

Out[197]:
```
count    0.0
mean     NaN
std      NaN
min      NaN
25%      NaN
50%      NaN
75%      NaN
max      NaN
Name: unit_order_patrol, dtype: float64
```

In [198]:
```
players_feature_16.describe()
```

Out[198]:
```
count    0.0
mean     NaN
std      NaN
min      NaN
25%      NaN
50%      NaN
75%      NaN
max      NaN
Name: unit_order_vector_target_position, dtype: float64
```

In [199]:
```
players_feature_17.describe()
```

Out[199]:
```
count    0.0
mean     NaN
std      NaN
min      NaN
25%      NaN
50%      NaN
75%      NaN
max      NaN
Name: unit_order_radar, dtype: float64
```

In [200]:
```
players_feature_18.describe()
```

Out[200]:
```
count    0.0
mean     NaN
std      NaN
min      NaN
25%      NaN
50%      NaN
75%      NaN
max      NaN
Name: unit_order_set_item_combine_lock, dtype: float64
```

In [205]:
```
players_feature_19.describe()
```

Out[205]:
```
count    0.0
mean     NaN
std      NaN
min      NaN
25%      NaN
50%      NaN
75%      NaN
max      NaN
Name: unit_order_continue, dtype: float64
```

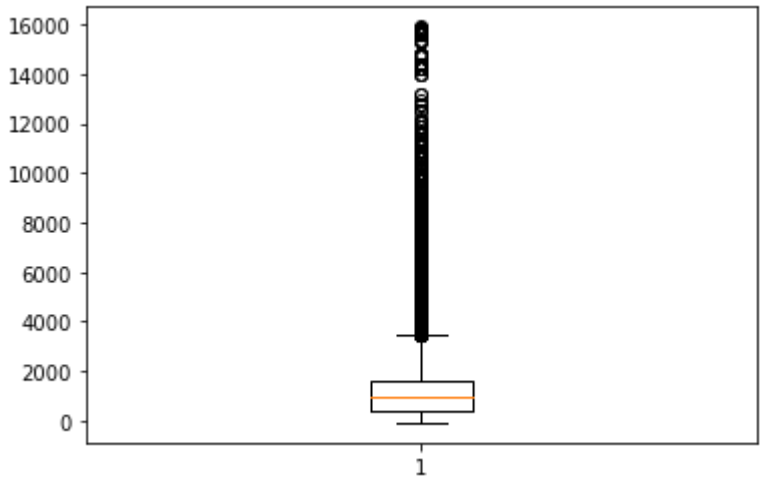
In [206]:
```
purchase_log

purchase_log_feature_1 = purchase_log["item_id"]
purchase_log_feature_2 = purchase_log["time"]
purchase_log_feature_3 = purchase_log["player_slot"]
purchase_log_feature_4 = purchase_log["match_id"]

purchase_log_feature_1.describe()
```

Out[206]:
```
count    1.819374e+07
mean     5.697155e+01
std      5.394464e+01
min      1.000000e+00
25%      2.500000e+01
50%      4.400000e+01
75%      5.100000e+01
max      2.540000e+02
Name: item_id, dtype: float64
```

In [207]:
```
purchase_log_feature_2.describe()
```

Out[207]:
```
count    1.819374e+07
mean     1.038303e+03
std      8.403476e+02
min     -9.000000e+01
25%      3.650000e+02
50%      9.200000e+02
75%      1.610000e+03
max      1.591400e+04
Name: time, dtype: float64
```

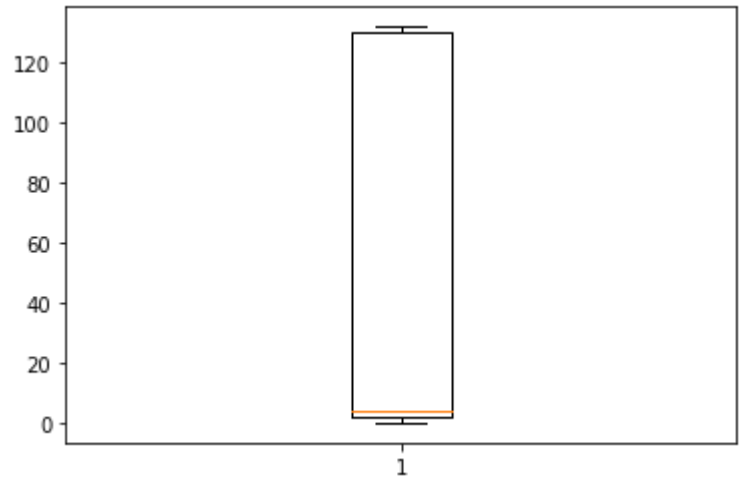
In [306]:
```
y = list(purchase_log.time)
plt.boxplot(y)
plt.show()
```



In [208]:
```
purchase_log_feature_3.describe()
```

Out[208]:
```
count    1.819374e+07
mean     6.591196e+01
std      6.401559e+01
min      0.000000e+00
25%      2.000000e+00
50%      4.000000e+00
75%      1.300000e+02
max      1.320000e+02
Name: player_slot, dtype: float64
```

In [307]:
```
y = list(purchase_log.player_slot)
plt.boxplot(y)
plt.show()
```




In [209]:
```
purchase_log_feature_4.describe()
```

Out[209]:
```
count    1.819374e+07
mean     2.501832e+04
std      1.443447e+04
min      0.000000e+00
25%      1.252900e+04
50%      2.504200e+04
75%      3.751800e+04
max      4.999900e+04
Name: match_id, dtype: float64
```

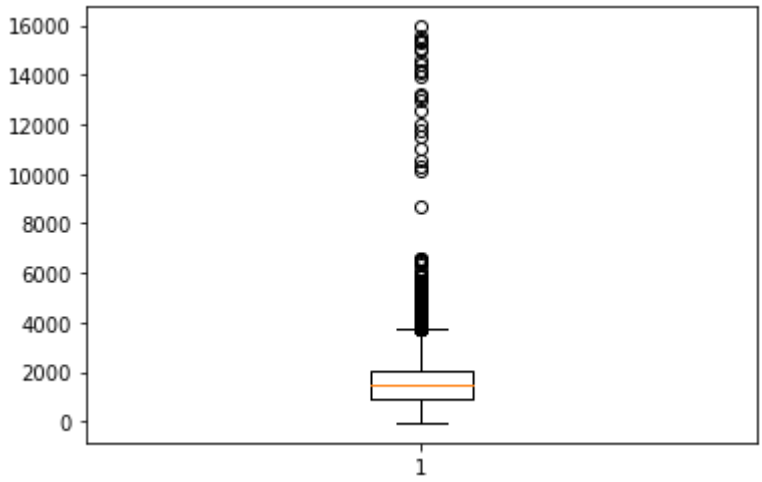
In [210]:
```
teamfights

teamfights_feature_1 = teamfights["match_id"]
teamfights_feature_2 = teamfights["start"]
teamfights_feature_3 = teamfights["end"]
teamfights_feature_4 = teamfights["last_death"]
teamfights_feature_5 = teamfights["deaths"]

teamfights_feature_1.describe()
```

Out[210]:
```
count    539047.000000
mean      24981.108779
std       14397.043269
min           0.000000
25%       12515.000000
50%       24994.000000
75%       37430.000000
max       49999.000000
Name: match_id, dtype: float64
```

In [211]:
```
teamfights_feature_2.describe()
```

Out[211]:
```
count    539047.000000
mean       1524.624868
std         779.104437
min         -49.000000
25%         923.000000
50%        1464.000000
75%        2039.000000
max       15941.000000
Name: start, dtype: float64
```

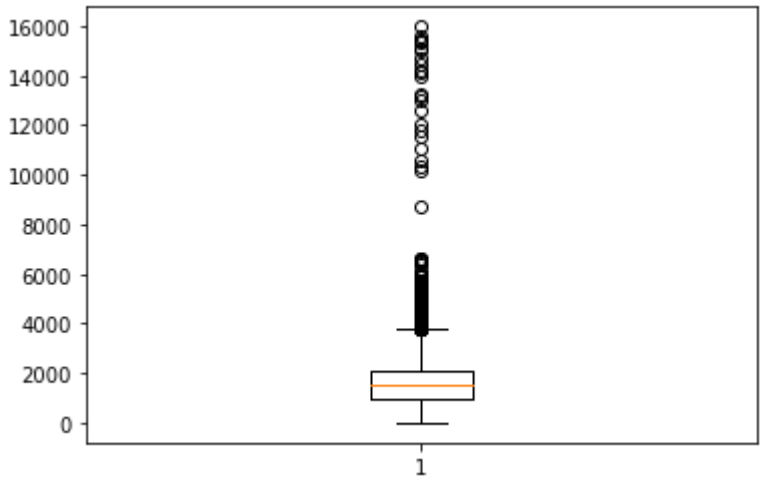
In [308]:
```
y = list(teamfights.start)
plt.boxplot(y)
plt.show()
```



In [212]:
```
teamfights_feature_3.describe()
```

Out[212]:
```
count    539047.000000
mean       1570.802991
std         779.974151
min          -1.000000
25%         968.000000
50%        1510.000000
75%        2086.000000
max       15979.000000
Name: end, dtype: float64
```

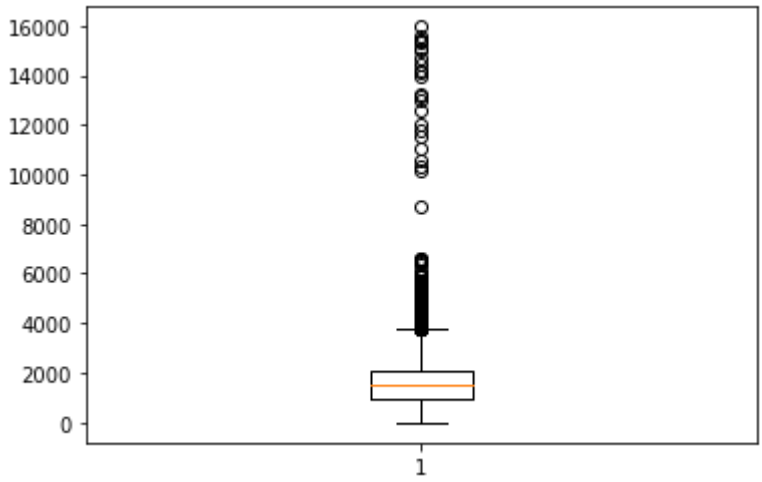
In [309]:
```
y = list(teamfights.end)
plt.boxplot(y)
plt.show()
```




In [213]:
```
teamfights_feature_4.describe()
```

Out[213]:
```
count    539047.000000
mean       1555.802991
std         779.974151
min         -16.000000
25%         953.000000
50%        1495.000000
75%        2071.000000
max       15964.000000
Name: last_death, dtype: float64
```

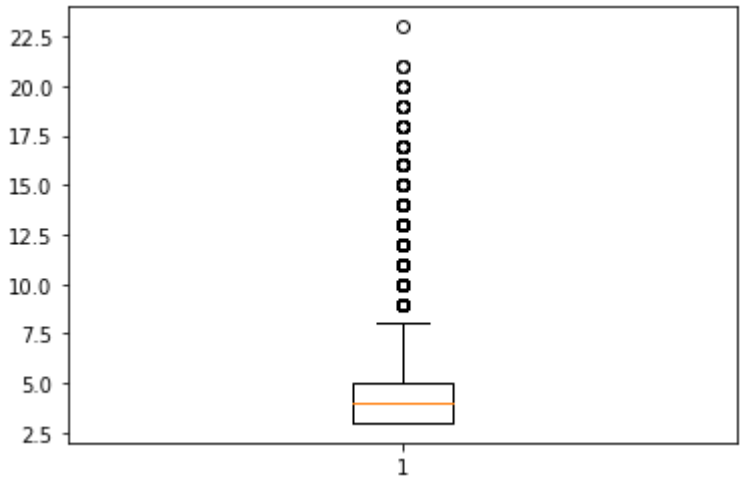
In [310]:
```
y = list(teamfights.last_death)
plt.boxplot(y)
plt.show()
```




In [214]:
```
teamfights_feature_5.describe()
```

Out[214]:
```
count    539047.000000
mean          4.324430
std           1.522701
min           3.000000
25%           3.000000
50%           4.000000
75%           5.000000
max          23.000000
Name: deaths, dtype: float64
```

In [311]:
```
y = list(teamfights.deaths)
plt.boxplot(y)
plt.show()
```



In [215]:
```
teamfights_players

teamfights_players_feature_1 = teamfights_players["match_id"]
teamfights_players_feature_2 = teamfights_players["player_slot"]
teamfights_players_feature_3 = teamfights_players["buybacks"]
teamfights_players_feature_4 = teamfights_players["damage"]
teamfights_players_feature_5 = teamfights_players["deaths"]
teamfights_players_feature_6 = teamfights_players["gold_delta"]
teamfights_players_feature_7 = teamfights_players["xp_end"]
teamfights_players_feature_8 = teamfights_players["xp_start"]

teamfights_players_feature_1.describe()

# test_labels
# test_player
```

Out[215]:
```
count    5.390470e+06
mean     2.498111e+04
std      1.439703e+04
min      0.000000e+00
25%      1.251500e+04
50%      2.499400e+04
75%      3.743000e+04
max      4.999900e+04
Name: match_id, dtype: float64
```

In [216]:
```
teamfights_players_feature_2.describe()
```

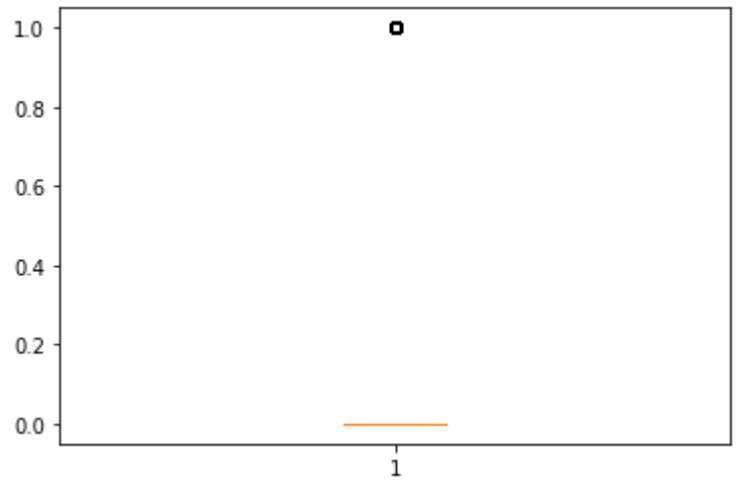
Out[216]:
```
count    5.390470e+06
mean     6.600000e+01
std      6.401563e+01
min      0.000000e+00
25%      2.000000e+00
50%      6.600000e+01
75%      1.300000e+02
max      1.320000e+02
Name: player_slot, dtype: float64
```

In [312]:
```
y = list(teamfights_players.player_slot)
plt.boxplot(y)
plt.show()
```


In [217]:
```
teamfights_players_feature_3.describe()
```

Out[217]:
```
count    5.390470e+06
mean     2.276147e-02
std      1.491422e-01
min      0.000000e+00
25%      0.000000e+00
50%      0.000000e+00
75%      0.000000e+00
max      1.000000e+00
Name: buybacks, dtype: float64
```

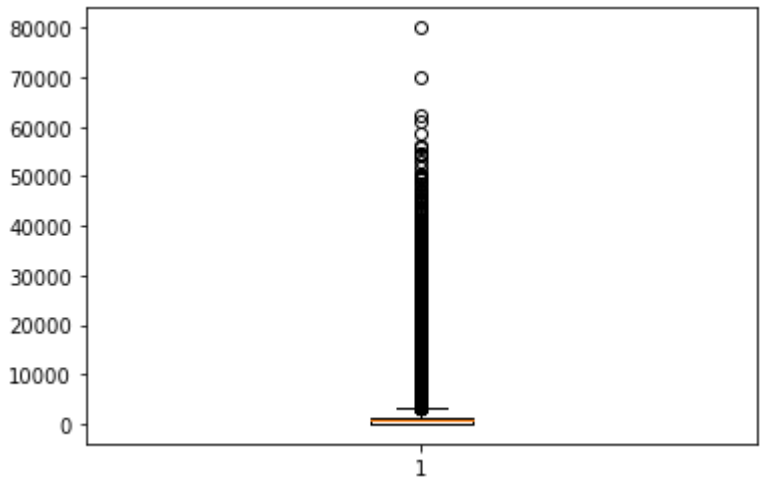
In [313]:
```
y = list(teamfights_players.buybacks)
plt.boxplot(y)
plt.show()
```



In [218]:
```
teamfights_players_feature_4.describe()
```

Out[218]:
```
count    5.390470e+06
mean     1.024086e+03
std      1.423389e+03
min      0.000000e+00
25%      1.480000e+02
50%      6.040000e+02
75%      1.353000e+03
max      8.001000e+04
Name: damage, dtype: float64
```

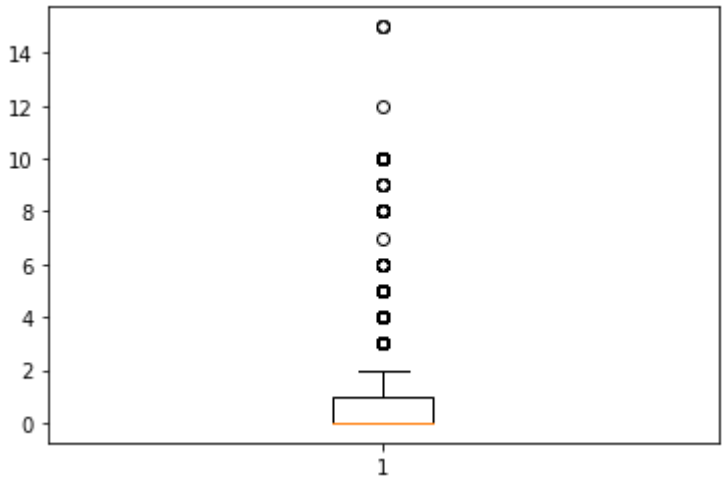
In [314]:
```
y = list(teamfights_players.damage)
plt.boxplot(y)
plt.show()
```




In [219]:
```
teamfights_players_feature_5.describe()
```

Out[219]:
```
count    5.390470e+06
mean     4.374474e-01
std      5.457010e-01
min      0.000000e+00
25%      0.000000e+00
50%      0.000000e+00
75%      1.000000e+00
max      1.500000e+01
Name: deaths, dtype: float64
```

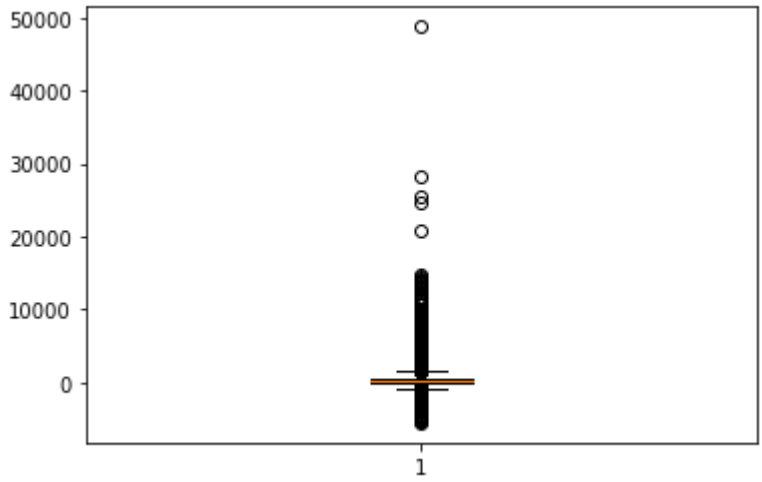
In [315]:
```
y = list(teamfights_players.deaths)
plt.boxplot(y)
plt.show()
```




In [220]:
```
teamfights_players_feature_6.describe()
```

Out[220]:
```
count    5.390470e+06
mean     2.580012e+02
std      5.787074e+02
min     -5.562000e+03
25%     -8.900000e+01
50%      1.660000e+02
75%      5.400000e+02
max      4.880000e+04
Name: gold_delta, dtype: float64
```

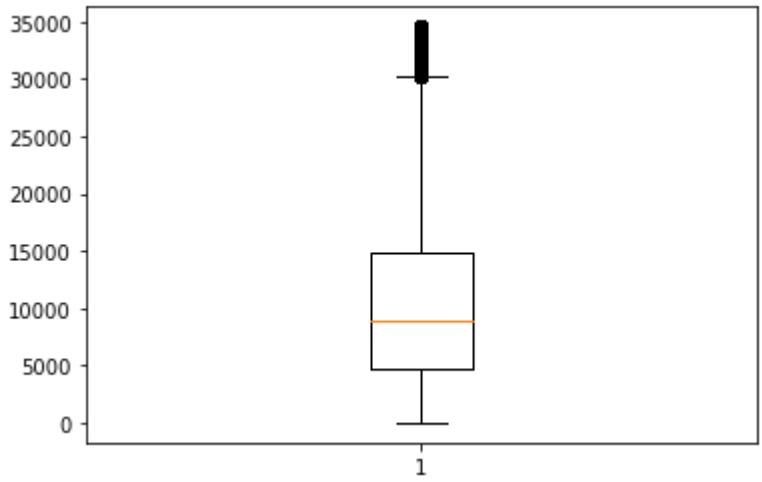
In [316]:
```
y = list(teamfights_players.gold_delta)
plt.boxplot(y)
plt.show()
```




In [221]:
```
teamfights_players_feature_7.describe()
```

Out[221]:
```
count    5.390470e+06
mean     1.006097e+04
std      7.491389e+03
min      0.000000e+00
25%      4.282000e+03
50%      8.207000e+03
75%      1.399200e+04
max      3.457400e+04
Name: xp_start, dtype: float64
```

In [317]:
```
y = list(teamfights_players.xp_end)
plt.boxplot(y)
plt.show()
```




In [221]:
```
teamfights_players_feature_7.describe()
```

Out[221]:
```
count    5.390470e+06
mean     1.006097e+04
std      7.491389e+03
min      0.000000e+00
25%      4.282000e+03
50%      8.207000e+03
75%      1.399200e+04
max      3.457400e+04
Name: xp_start, dtype: float64
```

In [ ]:
```
y = list(teamfights_players.xp_end)
plt.boxplot(y)
plt.show()
```

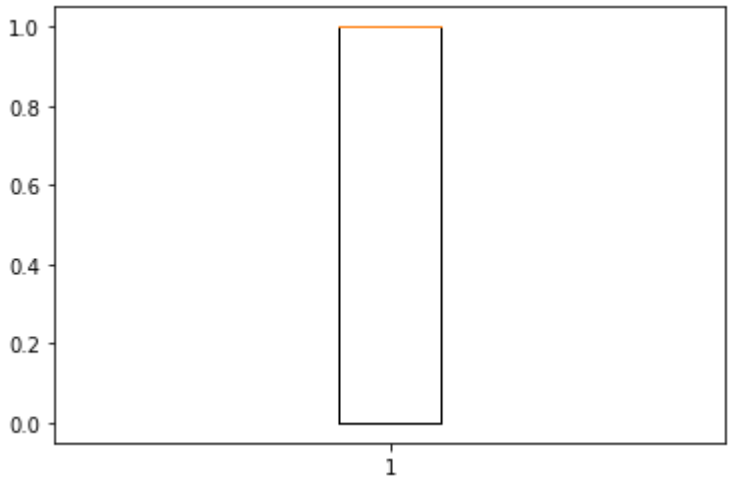
In [222]:
```
test_labels

test_labels_feature_1 = test_labels["match_id"]
test_labels_feature_2 = test_labels["radiant_win"]

test_labels_feature_1.describe()
```

Out[222]:
```
count    100000.000000
mean      99999.500000
std       28867.657797
min       50000.000000
25%       74999.750000
50%       99999.500000
75%      124999.250000
max      149999.000000
Name: match_id, dtype: float64
```

In [223]:
```
test_labels_feature_2.describe()
```

Out[223]:
```
count    100000.000000
mean          0.518610
std           0.499656
min           0.000000
25%           0.000000
50%           1.000000
75%           1.000000
max           1.000000
Name: radiant_win, dtype: float64
```

In [318]:
```
y = list(test_labels.radiant_win)
plt.boxplot(y)
plt.show()
```



In [230]:
```
test_player

test_player_feature_1 = test_player["match_id"]
test_player_feature_2 = test_player["account_id"]
test_player_feature_3 = test_player["hero_id"]
test_player_feature_4 = test_player["player_slot"]

test_player_feature_1.describe()
```

Out[230]:
```
count    1000000.000000
mean       99999.500000
std        28867.527892
min        50000.000000
25%        74999.750000
50%        99999.500000
75%       124999.250000
max       149999.000000
Name: match_id, dtype: float64
```

In [231]:
```
test_player_feature_2.describe()
```

Out[231]:
```
count    1000000.000000
mean       91028.072414
std       103566.804928
min            0.000000
25%            0.000000
50%        39462.500000
75%       177762.000000
max       330514.000000
Name: account_id, dtype: float64
```

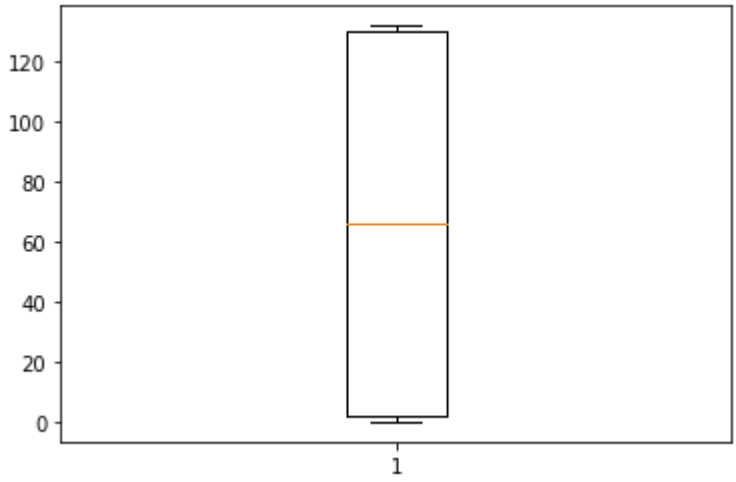
In [232]:
```
test_player_feature_3.describe()
```

Out[232]:
```
count    1000000.000000
mean          50.960134
std           33.229504
min            0.000000
25%           21.000000
50%           48.000000
75%           75.000000
max          112.000000
Name: hero_id, dtype: float64
```

In [233]:
```
test_player_feature_4.describe()
```

Out[233]:
```
count    1000000.000000
mean          66.000000
std           64.015655
min            0.000000
25%            2.000000
50%           66.000000
75%          130.000000
max          132.000000
Name: player_slot, dtype: float64
```

In [319]:
```
y = list(test_player.player_slot)
plt.boxplot(y)
plt.show()
```




In [ ]:
```

```